Acta Informatica manuscript No.
(will be inserted by the editor)

Trajectory-Based Codes

Michael Domaratzki*

School of Computing, Queen's University
Kingston, ON K7L 3N6 Canada
e-mail: domaratz@cs.queensu.ca

The date of receipt and acceptance will be inserted by the editor

Abstract. The notion of shuffle on trajectories is a natural generalization of many word operations considered in the literature. For a set of trajectories T, we define the notion of a T-code and examine its properties. Particular instances of T-codes are prefix-, suffix-, infix-, outfix- and hyper-codes, as well as other classes studied in the literature.

Key words. codes – shuffle on trajectories – convexity – binary relations

1 Introduction

The theory of codes is a fundamental area of formal language theory, with many important applications. The class of prefix codes is a particularly important subclass of codes, and is fundamentally linked to the nature of catenation as the underlying operation. Further research in codes has considered the subclasses of codes which arise from replacing catenation with other, related operations, most notably shuffle (the hypercodes) and insertion (the outfix codes).

In this paper, we generalize these results by considering T-codes. A T-code is any language L satisfying the equation $(L \sqcup\!\sqcup_T \Sigma^+) \cap L = \emptyset$, where $\sqcup\!\sqcup_T$ is a word operation defined by *shuffle on trajectories*. Shuffle on trajectories (see Section 2 for definitions) was defined by Mateescu *et al.* [35] and generalizes many word operations considered

* Research supported in part by an NSERC PGS-B graduate scholarship.

in the literature, including shuffle, insertion and concatenation. Thus, we consider the natural extension of codes to all operations defined by shuffle on trajectories, and examine the properties of these classes of languages.

The idea of studying general classes of codes has received much attention in the literature (see, e.g., Shyr and Thierrin [39], Jürgensen *et al.* [23] and Jürgensen and Yu [24]). Further, the definition of a T-code which we present can also be formulated in dependency theoretic terms (see, e.g., Jürgensen and Konstantinidis [22] for a survey of dependency theory). Some of the results we have obtained can be proven by appealing to dependency theory, however, our proofs are simpler in our restricted situation.

In addition, there are works in the literature which consider the problem of defining codes based on arbitrary binary relations, see, e.g., the work of Jürgensen *et al.* [23] on codes defined by binary relations and Shyr and Thierrin [39] for work on so-called *strict* binary relations. We will see that we can also view T-codes as anti-chains under the natural binary relation defined by T.

With this research in mind, we nonetheless feel the framework of T-codes is useful in that it helps us to see results relating to codes defined by shuffle on trajectories in a new way. The restriction of considering only those codes defined by shuffle on trajectories gives us new insight into these classes, including prefix-, suffix-, bi(pre)fix-, infix-, outfix-, shuffle- and hyper-codes, by focusing our attention to classes of codes which are specific enough to allow reasoning on the associated sets of trajectories, but general enough to encompass all of the above interesting and much-studied classes of codes.

We also feel that introducing the notion of T-code will allow more unified results to be obtained on the various classes of codes, since specific conditions on sets of trajectories (i.e., languages) will be easier to obtain than more general conditions on arbitrary relations. In particular, we have obtained results which do not appear to have been considered before in the more general framework of dependency theory or binary relations.

Further, we note that the idea of T-codes is useful elsewhere in the study of iterated shuffle and deletion along trajectories, for instance, in analyzing the *shuffle-base* of certain languages. Finally, the study of T-codes, much like the study of shuffle on trajectories in general, allows us to examine what assumptions must be made on an operation in order for certain results to follow. We find that even when these assumptions have been studied in the literature, the proofs obtained for the specific cases of shuffle on trajectories are often simpler.

We obtain several interesting results on T-codes. We generalize a result relating outfix and hyper-codes and the notion of (embedding-) convexity to all T-codes. Further, the known closure properties of shuffle on trajectories allow us to easily conclude positive decidability results for the problem of determining membership in classes of T-codes (including maximal T-codes), which were previously determined by ad-hoc constructions in the literature.

We note that recently, a more general concept than T-codes has been independently introduced by Kari *et al.* [27], motivated by the bonding of strands of DNA and DNA computing. Their framework, called *bond-free properties*, is also a general setting which involves shuffle on trajectories. Generally, the motivations for our work and those of the work by Kari *et al.* are different, and the decidability results which are similar are noted below.

2 Definitions

For additional background in formal languages and automata theory, please see Yu [44] or Hopcroft and Ullman [16]. Let Σ be a finite set of symbols, called *letters*. Then Σ^* is the set of all finite sequences of letters from Σ, which are called *words*. The empty word ϵ is the empty sequence of letters. The *length* of a word $w = w_1 w_2 \cdots w_n \in \Sigma^*$, where $w_i \in \Sigma$, is n, and is denoted $|w|$. Note that ϵ is the unique word of length 0. A *language* L is any subset of Σ^*. By $\overline{L}$, we mean $\Sigma^* - L$, the complement of L. If $L_1, \ldots, L_k \subseteq \Sigma^*$ are languages, we use the notation $\prod_{i=1}^{k} L_i = L_1 L_2 \cdots L_k$. If L is a language and k is a natural number, then we denote $L^{\leq k} = \{u_1 u_2 \cdots u_i \ : \ i \leq k, u_j \in L \ \forall 1 \leq j \leq i\}$.

We refer the reader to Rozenberg and Salomaa [36] for the definitions of the regular, linear context-free, context-free and recursive languages; these are denoted by REG, LCF, CF and REC, respectively.

We denote by $\mathbb{N}$ the set of natural numbers: $\mathbb{N} = \{0, 1, 2, \ldots\}$. If we wish to refer to the positive numbers, we will use the notation $\mathbb{N}^+ = \{1, 2, \ldots, \}$. Let $I \subseteq \mathbb{N}$. If there exist $n_0, p \in \mathbb{N}$, $p > 0$, such that for all $i \geq n_0$, $i \in I \iff i + p \in I$, then we say that I is *ultimately periodic*. For $n, m \in \mathbb{N}$, we use the notation $m \mid n$ to denote that m is a divisor of n, that is, there exists $k \in \mathbb{N}$ such that $n = km$.

Given alphabets Σ, Δ, a *morphism* is a function $h : \Sigma^* \to \Delta^*$ satisfying $h(xy) = h(x)h(y)$ for all $x, y \in \Sigma^*$. Given a morphism $h : \Sigma^* \to \Delta^*$ and a language $L \subseteq \Sigma^*$, then the image of L under h is given by $h(L) = \{h(x) \ : \ x \in L\}$, while if $L' \subseteq \Delta^*$, the inverse image of L' under h is defined by $h^{-1}(L') = \{x \in \Sigma^* \ : \ h(x) \in L'\}$.

Given an word $w \in \Sigma^*$ and $a \in \Sigma$, $|w|_a$ is the number of occurrences of a in w. For an alphabet $\Sigma = \{a_1, a_2, \ldots, a_n\}$ with a specified order $a_1 < a_2 < \cdots < a_n$, the *Parikh mapping* is given by $\Psi : \Sigma^* \to \mathbb{N}^n$, as follows:

$$\Psi(w) = (|w|_{a_i})_{i=1}^n.$$

It is extended to $\Psi : 2^{\Sigma^*} \to 2^{\mathbb{N}^n}$ as expected.

Recall that a language $L \subseteq \Sigma^*$ is *bounded* if there exist $k \in \mathbb{N}$ and $w_1, w_2, \ldots, w_k \in \Sigma^*$ such that $L \subseteq w_1^* w_2^* \cdots w_k^*$. If L is not bounded we say that it is *unbounded*.

The shuffle on trajectories operation is a method for specifying the ways in which two input words may be combined to form a result. Each trajectory $t \in \{0,1\}^*$ with $|t|_0 = n$ and $|t|_1 = m$ specifies the manner in which we can form the shuffle on trajectories of two words of length n (as the left input word) and m (as the right input word). The word resulting from the shuffle along t will have a letter from the left input word in position i if the i-th symbol of t is 0, and a letter from the right input word in position i if the i-th symbol of t is 1.

We now recall the formal definition of *shuffle on trajectories*, originally given by Mateescu *et al.* [35]. Shuffle on trajectories is defined by first defining the shuffle of two words x and y over an alphabet Σ on a trajectory $t \in \{0,1\}^*$. We denote the shuffle of x and y along trajectory t by $x \sqcup_t y$.

If $x = ax'$, $y = by'$ (with $a, b \in \Sigma$) and $t = et'$ (with $e \in \{0,1\}$), then

$$x \sqcup_{et'} y = \begin{cases} a(x' \sqcup_{t'} by') \text{ if } e = 0; \\ b(ax' \sqcup_{t'} y') \text{ if } e = 1. \end{cases}$$

If $x = ax'$ ($a \in \Sigma$), $y = \epsilon$ and $t = et'$ ($e \in \{0,1\}$), then

$$x \sqcup_{et'} \epsilon = \begin{cases} a(x' \sqcup_{t'} \epsilon) \text{ if } e = 0; \\ \emptyset \qquad\qquad \text{otherwise.} \end{cases}$$

If $x = \epsilon$, $y = by'$ ($b \in \Sigma$) and $t = et'$ ($e \in \{0,1\}$), then

$$\epsilon \sqcup_{et'} y = \begin{cases} b(\epsilon \sqcup_{t'} y') \text{ if } e = 1; \\ \emptyset \qquad\qquad \text{otherwise.} \end{cases}$$

We let $x \sqcup_\epsilon y = \emptyset$ if $\{x, y\} \neq \{\epsilon\}$. Finally, if $x = y = \epsilon$, then $\epsilon \sqcup_t \epsilon = \epsilon$ if $t = \epsilon$ and $\emptyset$ otherwise.

It is not difficult to see that if $t = \prod_{i=1}^{n} 0^{j_i}1^{k_i}$ for some $n \geq 0$ and $j_i, k_i \geq 0$ for all $1 \leq i \leq n$, then we have that

$$x \sqcup\!\sqcup_t y = \{\prod_{i=1}^{n} x_i y_i \ : \ x = \prod_{i=1}^{n} x_i, y = \prod_{i=1}^{n} y_i, \text{ with } |x_i| = j_i, |y_i| = k_i \text{ for all } 1 \leq i \leq n\}$$

if $|x| = |t|_0$ and $|y| = |t|_1$ and $x \sqcup\!\sqcup_t y = \emptyset$ if $|x| \neq |t|_0$ or $|y| \neq |t|_1$.

We extend shuffle on trajectories to sets $T \subseteq \{0,1\}^*$ of trajectories as follows:

$$x \sqcup\!\sqcup_T y = \bigcup_{t \in T} x \sqcup\!\sqcup_t y.$$

Further, for $L_1, L_2 \subseteq \Sigma^*$, we define

$$L_1 \sqcup\!\sqcup_T L_2 = \bigcup_{\substack{x \in L_1 \\ y \in L_2}} x \sqcup\!\sqcup_T y.$$

We will also require the following definition, introduced independently by the author [6] and Kari and Sosík [28], called *deletion along trajectories*, which models deletion operations controlled by a set of trajectories. Let $x, y \in \Sigma^*$ be words with $x = ax'$, $y = by'$ $(a, b \in \Sigma)$. Let t be a word over $\{i, d\}$ such that $t = et'$ with $e \in \{i, d\}$. Then we define $x \rightsquigarrow_t y$, the deletion of y from x along trajectory t, as follows:

$$x \rightsquigarrow_t y = \begin{cases} a(x' \rightsquigarrow_{t'} by') & \text{if } e = i; \\ x' \rightsquigarrow_{t'} y' & \text{if } e = d \text{ and } a = b; \\ \emptyset & \text{otherwise.} \end{cases}$$

Also, if $x = ax'$ $(a \in \Sigma)$ and $t = et'$ $(e \in \{i, d\})$, then

$$x \rightsquigarrow_t \epsilon = \begin{cases} a(x' \rightsquigarrow_{t'} \epsilon) & \text{if } e = i; \\ \emptyset & \text{otherwise.} \end{cases}$$

If $x \neq \epsilon$, then $x \rightsquigarrow_\epsilon y = \emptyset$. Further, $\epsilon \rightsquigarrow_t y = \epsilon$ if $t = y = \epsilon$. Otherwise, $\epsilon \rightsquigarrow_t y = \emptyset$.

Let $T \subseteq \{i, d\}^*$. Then

$$x \rightsquigarrow_T y = \bigcup_{t \in T} x \rightsquigarrow_t y.$$

We extend this to languages as expected: Let $L_1, L_2 \subseteq \Sigma^*$ and $T \subseteq \{i, d\}^*$. Then

$$L_1 \rightsquigarrow_T L_2 = \bigcup_{\substack{x \in L_1 \\ y \in L_2}} x \rightsquigarrow_T y.$$

We now come to the main definition of the paper. Let $L \subseteq \Sigma^+$ be a language. Then, for any $T \subseteq \{0,1\}^*$, we say that L is a *T-code* if L is non-empty and $(L \sqcup\!\sqcup_T \Sigma^+) \cap L = \emptyset$. If Σ is an alphabet and $T \subseteq \{0,1\}^*$, let $\mathcal{P}_T(\Sigma)$ denote the set of all T-codes over Σ. If Σ is understood, we will denote the set of T-codes over Σ by $\mathcal{P}_T$.

There has been much research into the idea of T-codes for particular $T \subseteq \{0,1\}^*$, including

(a) prefix codes, corresponding to $T = 0^*1^*$ (catenation);
(b) suffix codes, corresponding to $T = 1^*0^*$ (anti-catenation);
(c) biprefix (or bifix) codes, corresponding to $T = 0^*1^* + 0^*1^*$ (bi-catenation);
(d) outfix and infix codes, corresponding to $T = 0^*1^*0^*$ (insertion) and $T = 1^*0^*1^*$, (bi-polar insertion) respectively;
(e) shuffle-codes, corresponding to bounded trajectories such as
  (e-i) $T = (0^*1^*)^n$ for fixed $n \geq 1$ (prefix codes of index n);
  (e-ii) $T = (1^*0^*)^n$ for fixed $n \geq 1$ (suffix codes of index n);
  (e-iii) $T = 1^*(0^*1^*)^n$ for fixed $n \geq 1$ (infix codes of index n);
  (e-iv) $T = (0^*1^*)^n0^*$ for fixed $n \geq 1$ (outfix codes of index n);
(f) hypercodes, corresponding to $T = (0+1)^*$ (arbitrary shuffle); and
(g) k-codes, corresponding to $T = 0^*1^*0^{\leq k}$ (k-catenation, see Kari and Thierrin [29]) for fixed $k \geq 0$.
(h) for arbitrary $k \geq 1$, codes defined by the sets of trajectories $PP_k = 0^* + (0^*1^*)^{k-1}0^*1^+$, $PS_k = 0^* + 1^+0^*(1^*0^*)^{k-1}$, $PI_k = 0^* + (1^*0^*)^k1^+$, $SI_k = 0^* + 1^+(0^*1^*)^k$, $PB_k = PP_k \cup PS_k$ and $BI_k = PI_k \cup SI_k$, see Long [32], or Ito *et al.* [20] for PI_1, SI_1.

For a list of references related to (a)–(f), see Jürgensen and Konstantinidis [22, pp. 549–553]. In this paper, we let $H = (0+1)^*$, $P = 0^*1^*$, $S = 1^*0^*$, $I = 1^*0^*1^*$, $O = 0^*1^*0^*$ and $B = P + S$.

Recall that a non-empty language L is a *code* if $u_1u_2\cdots u_m = v_1v_2\cdots v_n$ where $u_i, v_j \in L$ for $1 \leq i \leq m$ and $1 \leq j \leq n$ implies that $n = m$ and $u_i = v_i$ for $1 \leq i \leq n$. For background on codes, we refer the reader to Berstel and Perrin [1], Jürgensen and Konstantinidis [22] or Shyr [37].

3 General Properties of T-codes

We can give two alternate characterizations of T-codes in terms of its left and right inverses (in the sense of Kari [26]). These are given via the morphisms $\tau, \pi : \{0,1\}^* \to \{i,d\}^*$ defined by $\tau(0) = i$, $\tau(1) = d$, $\pi(0) = d$ and $\pi(1) = i$. We can easily prove the following two equalities by appealing to results relating shuffle and deletion along

trajectories (see the work of the author [6] or Kari and Sosík [28]). In particular, we have for all $T \subseteq \{0,1\}^*$, and all Σ,

$$\mathcal{P}_T(\Sigma) = \{L \ : \ (L \rightsquigarrow_{\tau(T)} \Sigma^+) \cap L = \emptyset\}, \tag{1}$$
$$\mathcal{P}_T(\Sigma) = \{L \ : \ L \rightsquigarrow_{\pi(T)} L \subseteq \{\epsilon\}\}. \tag{2}$$

For some particular T, these characterizations are well-known, e.g., (1) for $T = 0^*1^*$ is given by Berstel and Perrin [1, Prop. II.1.1.(ii)].

We now note that the term T-code is somewhat of a misnomer: some T-codes are not codes. However, we feel that the use of the term appropriately encapsulates a sufficiently similar notion in terms of the language equation involved. In particular, note the following example:

Example 1. Let $T = (01)^*$. Then $\sqcup\!\sqcup_T$ corresponds to perfect shuffle (also known as balanced literal shuffle). Then note that $L = \{aa, bb, aabb\}$ is a T-code: there is no way to perfectly shuffle aa (resp., bb) and any other word of length 2 to get $aabb$. However, L is not a code: $aa \cdot bb = aabb$.

On the other hand, we shall see in Corollary 1 below that if $T \supseteq 0^*1^*$, then all T-codes are codes.

The following states that more restrictive sets of trajectories (potentially) result in more languages being T-codes; the proof is immediate:

Lemma 1. *Let $T_1 \subseteq T_2 \subseteq \{0,1\}^*$. Then for all Σ, $\mathcal{P}_{T_1}(\Sigma) \supseteq \mathcal{P}_{T_2}(\Sigma)$.*

By the fact that all prefix codes are codes, we conclude the following, which complements Example 1:

Corollary 1. *Let $T \supseteq 0^*1^*$. Then every T-code is a code.*

Let $\mathcal{P}_{\text{CODE}}$ denote the set of all codes. We now show that for all $T \subseteq \{0,1\}^*$, $\mathcal{P}_T \neq \mathcal{P}_{\text{CODE}}$. We will require the following well-known characterization of two element codes (see, e.g., Berstel and Perrin [1, Cor. 2.9]):

Theorem 1. *Let $L = \{x_1, x_2\} \subseteq \Sigma^+$. Then L is not a code iff there exist $z \in \Sigma^+$, $i, j \in \mathbb{N}^+$ such that $x_1 = z^i$ and $x_2 = z^j$.*

Lemma 2. *Let $T \subseteq \{0,1\}^*$. Then $\mathcal{P}_T(\Sigma) \neq \mathcal{P}_{\text{CODE}}(\Sigma)$ for all Σ with $|\Sigma| > 1$.*

Proof. Let $T \subseteq \{0,1\}^*$. If $T \subseteq 0^* + 1^*$, then $\mathcal{P}_T = \mathcal{P}_\emptyset = 2^{\Sigma^+} - \{\emptyset\}$ (the first equality will become clear after Theorem 3 below), which is clearly not the set of codes.

Thus, we can assume that there is some $t \in T$ with $|t|_1, |t|_0 > 0$. Let $n = |t|_0$. Consider that $t \in 0^n \sqcup\!\sqcup_t \{0,1\}^+$, thus $L = \{t, 0^n\}$ is not a T-code.

If L is not a code, then t and 0^n are powers of the same word, i.e., $t \in 0^*$. This contradicts our choice of t. Thus, L is a code. □

We also observe that $\mathcal{P}_{T_1} \cap \mathcal{P}_{T_2} = \mathcal{P}_{T_1 \cup T_2}$. We note that the dual case does not hold. In the case of $\mathcal{P}_{T_1 \cap T_2}$, we have the inclusion $\mathcal{P}_{T_1} \cap \mathcal{P}_{T_2} \subseteq \mathcal{P}_{T_1 \cap T_2}$. But of course equality does not hold in general. For example, with $T_1 = 0^*1^*$ and $T_2 = 1^*0^*$, $\mathcal{P}_{T_1 \cap T_2} = \mathcal{P}_{0^*+1^*} = \mathcal{P}_\emptyset = 2^{\Sigma^+} - \{\emptyset\}$ (the second equality will be established in Theorem 3 below). However, $\mathcal{P}_{T_1} \cap \mathcal{P}_{T_2} = \mathcal{P}_{T_1 \cup T_2}$, the set of biprefix codes.

We can also ask if $T_1 \subset T_2$ ($\subset$ denotes proper inclusion) implies that $\mathcal{P}_{T_1} \supset \mathcal{P}_{T_2}$. The answer is yes, as long as the difference between T_2 and T_1 contains non-unary words.

Theorem 2. *Let $T_1 \subset T_2$ be such that $(T_2 - T_1) \cap \overline{0^* + 1^*} \neq \emptyset$. Then for all Σ with $|\Sigma| \geq 2$, $\mathcal{P}_{T_1}(\Sigma) \supset \mathcal{P}_{T_2}(\Sigma)$.*

Proof. Let $t \in (T_2 - T_1) \cap \overline{0^* + 1^*}$. Let t_0, t_1 be defined by $t_0 = 0^{|t|_0}$ and $t_1 = 1^{|t|_1}$. Then note that $t_0, t_1 \neq \epsilon$, by our choice of t. Thus, we have that $\{t, t_0\} \subseteq \{0,1\}^+$. We claim that $L_t = \{t, t_0\} \in \mathcal{P}_{T_1} - \mathcal{P}_{T_2}$.

To see that $L_t \notin \mathcal{P}_{T_2}$, note that $t \in t_0 \sqcup\!\sqcup_t t_1$. As $t \in T_2$ and $t_1 \neq \epsilon$, L_t is not at T_2-code. Assume that L_t is not a T_1-code. As $|t| > |t_0|$, the only way that L_t can fail to be a T_1-code is if there exists $x \in \{0,1\}^+$ such that $t \in t_0 \sqcup\!\sqcup_{T_1} x$. By definition, this implies that $x = t_1$. But $t \in t_0 \sqcup\!\sqcup_{T_1} t_1$ only if $t \in T_1$, which is not the case. □

Theorem 3. *Let $T_1 \subset T_2$ and $T_2 - T_1 \subseteq 1^* + 0^*$. Then for all Σ with $|\Sigma| > 1$, $\mathcal{P}_{T_1}(\Sigma) = \mathcal{P}_{T_2}(\Sigma)$.*

Proof. Assume, contrary to what we want to prove, that $L \subseteq \Sigma^+$ is a T_1-code which is not a T_2-code. As L is not a T_2-code, there exist $x, z \in L$, $y \in \Sigma^+$ and $t \in T_2$ such that $z \in x \sqcup\!\sqcup_t y$. As L is a T_1-code, $z \notin x \sqcup\!\sqcup_{T_1} y$. Thus $t \notin T_1$. By assumption, this implies that $t \in 1^* + 0^*$.

If $t \in 1^*$, then by definition of $\sqcup\!\sqcup_T$, $z \in x \sqcup\!\sqcup_t y$ implies that $x = \epsilon$, contrary to our choice of L. If $t \in 0^*$, then by definition, $y = \epsilon$, contrary to our choice of y. In either case, we have arrived at a contradiction. □

Thus, we have completely characterized when a restriction in trajectories corresponds to an increase in languages which are codes. In particular, we note the following corollary:

Corollary 2. *Let $T_1, T_2 \subseteq \{0,1\}^*$ be regular sets of trajectories. Then it is decidable whether $\mathcal{P}_{T_1} = \mathcal{P}_{T_2}$.*

Proof. We note that $\mathcal{P}_{T_1} = \mathcal{P}_{T_2}$ iff $(T_1 - T_2) \cup (T_2 - T_1) \subseteq 0^* + 1^*$. Since T_1, T_2 are regular, so is $(T_1 - T_2) \cup (T_2 - T_1)$, and the inclusion is decidable. □

We now examine further questions of decidability.

Lemma 3. *Let $T \subseteq \{0,1\}^*$ be a fixed CF set of trajectories. Then given a regular language L, it is decidable whether L is a T-code.*

Proof. Since L is regular and T is a CFL, $L \sqcup\!\sqcup_T \Sigma^+$, and $(L \sqcup\!\sqcup_T \Sigma^+) \cap L$ are CFLs. Thus, we can test whether $(L \sqcup\!\sqcup_T \Sigma^+) \cap L = \emptyset$, which precisely defines L being a T-code. □

This result can also be proved using dependency theory. As every $T \subseteq \{0,1\}^*$ defines a 3-dependence system, and in particular every context-free T defines a dependence system whose associated support can be accepted by a 3-tape PDA, the problem of determining membership in $\mathcal{P}_T$ is decidable; see Jürgensen and Konstantinidis [22, Sect. 9] for details. Further, Kari *et al.* [27, Thm. 4.7] establish a similar decidability result in their framework of *bond-free properties*. When translated to our setting, it states that given T, R regular, we can decide if $R \in \mathcal{P}_T$.

A class of languages $\mathcal{C}$ is said to have *decidable membership problem* if, given $L \subseteq \Sigma^*$ with $L \in \mathcal{C}$, it is decidable whether $x \in L$ for an arbitrary $x \in \Sigma^*$. We have the following positive decidability result:

Lemma 4. *Let $\mathcal{C}$ be a class of languages with decidable membership. Let $T \subseteq \{0,1\}^*$ be a set of trajectories such that $T \in \mathcal{C}$. Then given a finite language F, it is decidable whether $F \in \mathcal{P}_T$.*

Proof. Let $F \subseteq \Sigma^+$ be a finite set. Let $n = \max\{|x| \ : \ x \in F\}$. Since membership in T is decidable, we can test all $t \in \{0,1\}^{\leq n}$ for membership in T. Thus, we can effectively compute $T^{\leq n} = T \cap \{0,1\}^{\leq n}$. It is easily observed that $F \cap (F \sqcup\!\sqcup_{T^{\leq n}} L) = F \cap (F \sqcup\!\sqcup_T L)$ for all L.

Since $F, T^{\leq n}, \Sigma^+$ are regular, we can test $F \cap (F \sqcup\!\sqcup_{T^{\leq n}} \Sigma^+) = \emptyset$. Thus, the result follows. □

We conclude with the following method of constructing a T-code from an arbitrary language.

Lemma 5. *Let $T \subseteq \{0,1\}^*$. Let $L \subseteq \Sigma^+$ be a non-empty language. Then $L_0 = L - (L \sqcup\!\sqcup_T \Sigma^+) \in \mathcal{P}_T(\Sigma)$.*

Proof. As $L_0 \subseteq L$ and $\sqcup\!\sqcup_T$ is a monotone operation, $(L_0 \sqcup\!\sqcup_T \Sigma^+) \subseteq (L \sqcup\!\sqcup_T \Sigma^+)$. Thus, $L_0 \cap (L_0 \sqcup\!\sqcup_T \Sigma^+) \subseteq L_0 \cap (L \sqcup\!\sqcup_T \Sigma^+)$ and $L_0 \cap (L \sqcup\!\sqcup_T \Sigma^+) = \emptyset$ by definition of L_0. $\square$

4 The Binary Relation defined by Trajectories

We can also define T-codes by appealing to a definition based on binary relations. In particular, for $T \subseteq \{0,1\}^*$, define ω_T as follows: for all $x, y \in \Sigma^*$,

$$x \,\omega_T\, y \iff y \in x \sqcup\!\sqcup_T \Sigma^*.$$

Then it is clear that L is a T-code iff L is an anti-chain under ω_T (i.e, $x, y \in L$ and $x \,\omega_T\, y$ implies $x = y$).

We note that the relation analogous to ω_T for infinite words and ω-trajectories was defined by Kadrie *et al.* [25], and its properties were briefly investigated. Kadrie *et al.* do not investigate the analogous relation with the same amount of detail as below and do not appear to be motivated by coding theory.

We now recall some of the properties of the binary relations ω_T that will be useful. The proofs may be found in the companion paper by the author [7]. In what follows, we will refer to T having a property P iff ω_T has property P.

Anti-symmetry

We note that ω_T always gives an anti-symmetric binary relation:

Lemma 6 [7]. *Let $T \subseteq \{0,1\}^*$. The relation ω_T is anti-symmetric.*

ST-Strictness

Shyr and Thierrin [39] define the concept of a *strict* binary relation. To avoid confusion with the concept of a *strict ordering* (see, e.g., Choffrut and Karhumäki [3, Sect. 7.1]), we will call a binary relation ρ on Σ^* *ST-strict* if it satisfies the following four properties:

(a) ρ is reflexive;
(b) ρ is positive (i.e., $\epsilon \,\rho\, u$ for all $u \in \Sigma^*$ [7]);
(c) for all $u, v \in \Sigma^*$, $u \,\rho\, v$ implies $|u| \leq |v|$;
(d) for all $u, v \in \Sigma^*$, $u \,\rho\, v$ and $|u| = |v|$ implies $u = v$.

Lemma 7 [7]. *Let $T \subseteq \{0,1\}^*$. Then T is ST-strict iff $0^* + 1^* \subseteq T$.*

Corollary 3. *Given a CF set $T \subseteq \{0,1\}^*$ of trajectories, it is decidable whether T is ST-strict.*

Corollary 4 [7]. *Let $T_1, T_2 \subseteq \{0,1\}^*$ be ST-strict. Then $\mathcal{P}_{T_1} = \mathcal{P}_{T_2}$ iff $T_1 = T_2$.*

Cancellativity

A binary relation ρ on Σ^* is said to be *left-cancellative* (resp., *right-cancellative*) if $uv\,\rho\,ux$ implies $v\,\rho\,x$ (resp., $vu\,\rho\,xu$ implies $v\,\rho\,x$) for all $u, v, x \in \Sigma^*$. The relation ρ is *cancellative* if it is both left- and right-cancellative.

Given $T \subseteq \{0,1\}^*$, we define two sets of trajectories, $s(T), p(T) \subseteq \{0,1\}^*$, as follows:

$$p(T) = \{t_1 1^j \ : \ t_1 t_2 \in T, 0 \leq j \leq |t_2|\},$$
$$s(T) = \{1^j t_2 \ : \ t_1 t_2 \in T, 0 \leq j \leq |t_1|\}.$$

Lemma 8 [7]. *Let $T \subseteq \{0,1\}^*$. Then T is left-cancellative (resp., right-cancellative) if $s(T) \subseteq T$ (resp., $p(T) \subseteq T$).*

Corollary 5 [7]. *Let $T \subseteq \{0,1\}^*$. If $s(T) \cup p(T) \subseteq T$, then T is cancellative.*

We now consider a condition of Jürgensen *et al.* [23]. Say that a binary relation ρ on Σ^* is *leviesque* if $uv\,\rho\,xy$ implies that $u\,\rho\,x$ or $v\,\rho\,y$, for all $u, v, x, y \in \Sigma^*$.

Lemma 9 [7]. *Let $T \subseteq \{0,1\}^*$. If $s(T) \cup p(T) \subseteq T$, then T is leviesque.*

Compatibility

Let ρ be a binary relation on Σ^*. Then we say that ρ is *left-compatible* (resp., *right-compatible*) if, for all $u, v, w \in \Sigma^*$, $u\,\rho\,v$ implies that $uw\,\rho\,vw$ (resp., $wu\,\rho\,wv$). If ρ is both left- and right-compatible, we say it is *compatible*.

Lemma 10 [7]. *Let $T \subseteq \{0,1\}^*$. Then T is right-compatible (resp., left-compatible) iff $T0^* \subseteq T$ (resp., $0^*T \subseteq T$).*

Corollary 6 [7]. *Let $T \subseteq \{0,1\}^*$. Then T is compatible iff $0^*T0^* \subseteq T$.*

Let $\mathcal{P}_P, \mathcal{P}_S, \mathcal{P}_O$ be the class of prefix, suffix and outfix codes. We can conclude the following corollary about positive T which satisfy compatibility conditions. Parts (a) and (b) of the following result have been established for all partial orders by Jürgensen *et al.* [23]; the proofs are immediate in our case:

Corollary 7. *Let $T \subseteq \{0,1\}^*$ be positive. Then the following hold:*

(a) if T is left-compatible, then $\mathcal{P}_T \subseteq \mathcal{P}_P$;
(b) if T is right-compatible, then $\mathcal{P}_T \subseteq \mathcal{P}_S$;
(c) if T is compatible, then $\mathcal{P}_T \subseteq \mathcal{P}_O$.

Furthermore, in each case equality of the classes holds iff it holds for the sets of trajectories involved.

Proof. We prove (b); the rest are similar. If T is positive then $1^* \subseteq T$. If T is right compatible, then $T0^* \subseteq T$. Thus, $S = 1^*0^* \subseteq T$. The inclusions thus hold by Lemma 1; for the equalities, we note that P, S, O are ST-strict and for each of (a),(b) and (c), T is also ST-strict. □

Transitivity

We now consider conditions on T which will ensure that ω_T is a transitive relation. Transitivity is often, but not always, a property of the binary relations defining the classic code classes. For instance, both bi-prefix and outfix codes are defined by binary relations which are not transitive, and hence not a partial order.

Consider that if $\{T_i\}_{i \in I}$ is a family of transitive sets of trajectories, then set $\cap_{i \in I} T_i$ is also transitive. Thus, we can define the transitive closure of a set T of trajectories as follows: for all $T \subseteq \{0,1\}^*$, let $tr(T) = \{T' \subseteq \{0,1\}^* \ : \ T \subseteq T', T' \text{ transitive}\}$. Note that $tr(T) \neq \emptyset$, as $\{0,1\}^* \in tr(T)$ for all $T \subseteq \{0,1\}^*$. Define $\hat{T}$ as

$$\hat{T} = \bigcap_{T' \in tr(T)} T'.$$

Then note that $\hat{T}$ is transitive and is the smallest transitive set of trajectories containing T. The operation $\hat{\cdot} : 2^{\{0,1\}^*} \to 2^{\{0,1\}^*}$ is indeed a closure operator (much like the closure operators on sets of trajectories constructed by Mateescu *et al.* [35] for, e.g., associativity and commutativity) in the algebraic sense, since $T \subseteq \hat{T}$, and $\hat{\cdot}$ preserves inclusion and is idempotent. Thus, we can, for instance, note the following result:

Lemma 11. *If $T \supseteq O$ $(= 0^*1^*0^*)$, then $\hat{T} = H$ $(= \{0,1\}^*)$.*

Proof. The result follows, since it is known (and easily observed) that $\hat{O} = H$ (see, e.g., Ito *et al.* [20, Rem. 3.2]). □

For particular instances of Lemma 11, see Thierrin and Yu [42, Prop. 2.3] or Long [33, Thm. 2.1].

Let $D = \{x, y, z\}$ and $\varphi, \sigma, \psi : D^* \to \{0,1\}^*$ be the morphisms given by

$$\begin{array}{l} \varphi(x) = 0,\ \sigma(x) = 0,\ \psi(x) = 0, \\ \varphi(y) = 0,\ \sigma(y) = 1,\ \psi(y) = 1, \\ \varphi(z) = 1,\ \sigma(z) = \epsilon,\ \psi(z) = 1. \end{array}$$

Consider the operator $\Omega_T : 2^{\{0,1\}^*} \to 2^{\{0,1\}^*}$ given by

$$\Omega_T(T') = T \cup T' \cup \psi(\sigma^{-1}(T') \cap \varphi^{-1}(T')). \tag{3}$$

Then the following equality holds [7]:

$$\hat{T} = \bigcup_{i \geq 0} \Omega_T^i(T). \tag{4}$$

Monotonicity

A binary relation ρ on Σ^* is said to be *monotone* (see, e.g, Ehrenfeucht *et al.* [8, p. 315]) if $x \rho y$ and $u \rho v$ implies $xu \rho yv$ for all $x, y, u, v \in \Sigma^*$. Occasionally, the concept of monotonicity is included as a requirement in compatibility, but we separate the two concepts here for clarity. We note that monotone here is a condition on T, rather than the monotonicity of the operation $\sqcup_T$ (i.e., that $L_1 \subseteq L_2, L_3 \subseteq L_4$, and $T_1 \subseteq T_2$ imply that $L_1 \sqcup_{T_1} L_3 \subseteq L_2 \sqcup_{T_2} L_4$), which holds for all T.

Lemma 12 [7]. *Let $T \subseteq \{0,1\}^*$. Then T is monotone iff $T^2 \subseteq T$ iff $T = T^*$.*

Recall that $B = 0^*1^* + 1^*0^*$ and $\mathcal{P}_B$ corresponds to the set of biprefix codes.

Lemma 13. *Let $T \subseteq \{0,1\}^*$. If T is ST-strict and monotone, then $\mathcal{P}_T \subseteq \mathcal{P}_B$. Further, equality between $\mathcal{P}_T$ and $\mathcal{P}_B$ holds iff $T = B$.*

5 Convexity and Transitivity

Let $\hat{T}$ again represent the transitive closure of T. We now examine the relationship between T-codes and $\hat{T}$-codes for arbitrary $T \subseteq \{0,1\}^*$.

We call a language $L \subseteq \Sigma^*$ *T-convex* if, for all $y \in \Sigma^*$ and $x, z \in L$, $x\ \omega_T\ y$ and $y\ \omega_T\ z$ implies $y \in L$. The notion of T-convexity was considered by the author in a companion paper [7].

Theorem 4. *Let Σ be an alphabet and $T \subseteq \{0,1\}^*$. For all languages $L \subseteq \Sigma^+$, the following two conditions are equivalent:*

(i) L is a $\hat{T}$-code;
(ii) L is a $\hat{T}$-convex T-code.

Proof. (i) $\Rightarrow$ (ii): Let $L \subseteq \Sigma^+$ be a $\hat{T}$-code. Then as $T \subseteq \hat{T}$, L is a T-code as well. Assume that $u\ \omega_{\hat{T}}\ v\ \omega_{\hat{T}}\ w$, with $u, w \in L$. As $\hat{T}$ is transitive, by definition, $u\ \omega_{\hat{T}}\ w$. Thus, $u = w$, as $u, w \in L$. Now, by the antisymmetry of $\hat{T}$, $v\ \omega_{\hat{T}}\ w = u$ and $u\ \omega_{\hat{T}}\ v$ imply $v = u \in L$. Thus, L is $\hat{T}$-convex.
(ii) $\Rightarrow$ (i): Let $L \subseteq \Sigma^+$ be a T-code, as well as being $\hat{T}$-convex.

Recall the operator Ω_T given by (3). Let $T_i = \Omega_T^i(T)$. Then $\hat{T} = \cup_{i \geq 0} T_i$, by (4). We establish (by induction) that L is a T_i-code for all $i \geq 0$. The result will then follow. To see this, assume L is a T_i-code for all $i \geq 0$. Let $x, y \in L$ be such that $x\, \omega_{\hat{T}}\, y$. Then there exists $t \in \hat{T}$ such that $y \in x \sqcup\!\sqcup_t z$ for some $z \in \Sigma^*$. As $t \in \hat{T}$, there exists $i \geq 0$ such that $t \in T_i$, Thus, $x\ \omega_{T_i}\ y$ and $x = y$, as required.

We now establish by induction on $i \geq 0$ that L is a T_i-code. For $i = 0$, $T_0 = T$. Thus, L is a T-code by assumption.

Let $i > 0$ and assume that L is a T_{i-1}-code. Let $x, y \in L$ be chosen so that $x\ \omega_{T_i}\ y$. Thus, there exist $t \in T_i$ and $z \in \Sigma^*$ such that $y \in x \sqcup\!\sqcup_t z$. We have that $t \in T_i = \Omega_T(T_{i-1}) = T \cup T_{i-1} \cup \psi(\sigma^{-1}(T_{i-1}) \cap \varphi^{-1}(T_{i-1}))$. If $t \in T \cup T_{i-1}$, then, as $y \in x \sqcup\!\sqcup_t z$, by induction $x = y$.

Consider then the case when $t \in \psi(\sigma^{-1}(T_{i-1}) \cap \varphi^{-1}(T_{i-1}))$. Let $t_0, t_1 \in T_{i-1}$ be such that $t \in \psi(\sigma^{-1}(t_1) \cap \varphi^{-1}(t_0))$. By definition of ψ, σ, φ, we know that we can write

$$t_0 = \prod_{k=1}^{n} 0^{i_k} 1^{j_k}$$

for some $n \in \mathbb{N}$ and $i_k, j_k \in \mathbb{N}$ for all $1 \leq k \leq n$, as well as $t_1 = \prod_{k=1}^{n} s_k$ where $|s_k| = i_k$ for all $1 \leq k \leq n$. Further,

$$t = \prod_{k=1}^{n} s_k 1^{j_k}.$$

As $y \in x \sqcup\!\sqcup_t z$, we can write $x = \prod_{k=1}^{n} x_k$, $z = \prod_{k=1}^{n} \alpha_k\beta_k$, where $x_k, \alpha_k, \beta_k \in \Sigma^*$ satisfy $|x_k| = |s_k|_0$, $|\alpha_k| = |s_k|_1$ and $|\beta_k| = j_k$ for all $1 \leq k \leq n$. Further, let $y = \prod_{k=1}^{n} \gamma_k\beta_k$ where $\gamma_k \in x_k \sqcup\!\sqcup_{s_k} \alpha_k$ for all $1 \leq k \leq n$.

Let $\alpha = \prod_{k=1}^{n} \alpha_k$, $\beta = \prod_{k=1}^{n} \beta_k$ and $\gamma = \prod_{k=1}^{n} \gamma_k$. Then we note that

$$y \in \gamma \sqcup\!\sqcup_{t_0} \beta;$$
$$\gamma \in x \sqcup\!\sqcup_{t_1} \alpha.$$

As $t_0, t_1 \in T_{i-1} \subseteq \hat{T}$, we conclude that $x\, \omega_{T_{i-1}}\, \gamma\, \omega_{T_{i-1}}\, y$, as well as $x\, \omega_{\hat{T}}\, \gamma\, \omega_{\hat{T}}\, y$, and thus $\gamma \in L$, by the $\hat{T}$-convexity of L.

Finally, we note that $\gamma\, \omega_{T_{i-1}}\, y$ implies that $\gamma = y$, as L is a T_{i-1}-code by induction. Similarly, $x\, \omega_{T_{i-1}}\, \gamma$ implies that $\gamma = x$. We conclude that $x = y$ and, since $x, y \in L$ were chosen arbitrarily, L is a T_i-code. □

Theorem 4 was known for the case $O = 0^*1^*0^*$, which corresponds to outfix codes, see, e.g., Shyr and Thierrin [38, Prop. 2]. In this case, $\hat{O} = H = (0 + 1)^*$, which corresponds to hypercodes. Theorem 4 was known to Guo *et al.* [11, Prop. 2] in a slightly weaker form for $B = 0^*1^* + 1^*0^*$. In this case, $\hat{B} = I = 1^*0^*1^*$, and the convexity is with respect to the factor (or subword) ordering. See also Long [33, Sect. 5] for the case of shuffle codes.

6 Closure Properties

We now consider the closure properties of $\mathcal{P}_T$.

We note immediately that $\mathcal{P}_T$ is closed under intersection with arbitrary languages, provided the intersection is non-empty. Further, it is clear that $\mathcal{P}_T$ is closed under union only if $T \subseteq 0^* + 1^*$.

6.1 Closure under Catenation

Theorem 5. *Let $T \subseteq \{0,1\}^*$ be a set of trajectories such that $s(T) \cup p(T) \subseteq T$. Then $\mathcal{P}_T$ is closed under catenation.*

Proof. Let $L_i \in \mathcal{P}_T$ for $i = 1, 2$. Assume that

$$(L_1L_2 \sqcup\!\sqcup_T x) \cap L_1L_2 \neq \emptyset$$

for some $x \in \Sigma^*$. We will demonstrate that $x = \epsilon$. Let $\alpha_i, \beta_i \in L_i$ for $i = 1, 2$ be such that

$$\beta_1\beta_2 \in \alpha_1\alpha_2 \sqcup\!\sqcup_T x.$$

Let $t \in T$ be such that $\beta_1\beta_2 \in \alpha_1\alpha_2 \sqcup\!\sqcup_t x$. Let $x = x_1x_2$ and $t = t_1t_2$ be chosen so that $\beta_1\beta_2 \in (\alpha_1 \sqcup\!\sqcup_{t_1} x_1)(\alpha_2 \sqcup\!\sqcup_{t_2} x_2)$. We distinguish two cases:

(a) $|\alpha_1| + |x_1| \geq |\beta_1|$. Then there exists $\gamma \in \Sigma^*$ such that

$$\beta_1\gamma \in \alpha_1 \sqcup\!\sqcup_{t_1} x_1,$$
$$\beta_2 \in \gamma(\alpha_2 \sqcup\!\sqcup_{t_2} x_2).$$

Let $t_2' = 1^{|\gamma|}t_2$ and $x_2' = \gamma x_2$. Then, as $|\gamma| \leq |t_1|$, $t_2' \in s(T) \subseteq T$ and thus $\beta_2 \in \alpha_2 \sqcup\!\sqcup_{t_2'} x_2'$ implies that $x_2' = \epsilon$. In particular, $x_2 = \gamma = \epsilon$. As $\gamma = \epsilon$, $\beta_1 \in \alpha_1 \sqcup\!\sqcup_{t_1} x_1$. Note that $t_1 \in p(T) \subseteq T$. Thus, L_1 a T-code implies that $x_1 = \epsilon$ and hence $x = x_1x_2 = \epsilon$.

(b) $|\alpha_1| + |x_1| < |\beta_1|$. Let $\gamma \in \Sigma^+$ be such that

$$\beta_1 \in (\alpha_1 \sqcup\!\sqcup_{t_1} x_1)\gamma;$$
$$\gamma\beta_2 \in \alpha_2 \sqcup\!\sqcup_{t_2} x_2.$$

Let $t_1' = t_1 1^{|\gamma|} \in p(T) \subseteq T$, as $|\gamma| \leq |t_2|$, and let $x_1' = x_1\gamma$. Then $\beta_1 \in (\alpha_1 \sqcup\!\sqcup_{t_1'} x_1')$. As L_1 is a T-code, $x_1' = \epsilon$. This contradicts that $\gamma \in \Sigma^+$.

Thus, $x = \epsilon$ and L_1L_2 is a T-code. □

We note that Theorem 5 can also be proven as follows: as $p(T) \cup s(T) \subseteq T$, T is both cancellative and leviesque. By Jürgensen *et al.* [23, Prop. 10], this implies that $\mathcal{P}_T$ is closed under catenation.

6.2 Closure under Inverse Morphism

We now turn to inverse morphism. Let $n \geq 1$. Let $T \subseteq (0^*1^*)^n$ be a bounded regular language such that there exist a_i, b_i, c_i, d_i for $1 \leq i \leq n$ such that

$$T = \prod_{i=1}^{n} 0^{a_i}(0^{b_i})^*1^{c_i}(1^{d_i})^*. \tag{5}$$

(We assume throughout that $T \subseteq (0^*1^*)^n$; similar proofs follow if, e.g., $T \subseteq (0^*1^*)^n0^*$). Let

$$I_j = \{a_j + b_jm : m \geq 0\} \quad \forall 1 \leq j \leq n;$$
$$K_j = \{c_j + d_jm : m \geq 0\} \quad \forall 1 \leq j \leq n.$$

Let $I_j' = I_j \setminus \{0\}$ for all $1 \leq j \leq n$.

Let $\varphi : \Delta^* \to \Sigma^*$ be a morphism. We define $[\varphi], [\varphi^{-1}] : \mathbb{N} \to 2^{\mathbb{N}}$ as follows:

$$[\varphi](m) = \{|x| \ : \ x \in \varphi(\Sigma^m)\};$$
$$[\varphi^{-1}](m) = \{|x| \ : \ x \in \varphi^{-1}(\Sigma^m)\}.$$

We extend these functions naturally to operate on $2^{\mathbb{N}}$ as, e.g., $[\varphi](S) = \bigcup_{s \in S} [\varphi](s)$.

We now prove a generalization of a result on infix and outfix codes established by Ito *et al.* [20, Prop. 6.5].

Theorem 6. *Let $T \subseteq (0^*1^*)^n$ be a bounded regular set of trajectories as given by (5). Let $\varphi : \Delta^* \to \Sigma^*$ be a morphism satisfying*

(a) $\emptyset \neq [\varphi^{-1}](I_j) \subseteq I_j$ for all $1 \leq j \leq n$.
(b) there exists j with $1 \leq j \leq n$ such that $\emptyset \neq [\varphi^{-1}](I'_j) \subseteq I'_j$.
(c) $[\varphi](I_j) \subseteq I_j$ for all $1 \leq j \leq n$.
(d) $[\varphi](K_j) \subseteq K_j$ for all $1 \leq j \leq n$.

Then $\mathcal{P}_T$ is closed under φ^{-1} iff

$$\{|x| \ : \ x \in \varphi^{-1}(\epsilon)\}^n \cap \left(\prod_{j=1}^{n} K_j - \{0\}^n \right) = \emptyset. \tag{6}$$

Proof. Assume that (6) fails. Let x_j for $1 \leq j \leq n$ be such that $x_j \in \varphi^{-1}(\epsilon)$ and $|x_j| \in K_j$. By (6), $x = \prod_{i=1}^{n} x_i \neq \epsilon$. Let $k_j = |x_j|$ for $1 \leq j \leq n$.

By (a), let $i_j \in I_j$ be such that $[\varphi^{-1}](i_j) \neq \emptyset$ for all $1 \leq j \leq n$, and such that there exists j_0 satisfying $1 \leq j_0 \leq n$, $i_{j_0} \neq 0$ and $[\varphi^{-1}](i_{j_0})$ contains a non-zero element, by (b). Thus, $\varphi^{-1}(\Sigma^{i_j}) \neq \emptyset$. Let $u_j \in \Sigma^{i_j}$ be such that there exist $v_j \in \varphi^{-1}(u_j)$ for all $1 \leq j \leq n$. As $i_{j_0} \neq 0$, $u = \prod_{j=1}^{n} u_j \neq \epsilon$, and as we can choose $v_{j_0} \in \varphi^{-1}(u_{j_0})$ to be a non-empty word, $v = \prod_{j=1}^{n} v_j \neq \epsilon$. Further, by (a), $|v_j| \in I_j$. Let $\ell_j = |v_j|$ for $1 \leq j \leq n$.

Consider $t = \prod_{j=1}^{n} 0^{\ell_j} 1^{k_j}$. As $\ell_j \in I_j$ and $k_j \in K_j$, $t \in T$. We now define a T-code $L \subseteq \Sigma^+$ such that $\varphi^{-1}(L)$ is not a T-code.

Consider $L = \{u\} \subseteq \Sigma^+$. Trivially, L is a T-code. Let $w = \prod_{j=1}^{n} v_j x_j$. Note that $\varphi(v) = \varphi(v_1) \cdots \varphi(v_n) = u_1 \cdots u_n = u$, and that $\varphi(w) = \prod_{j=1}^{n} \varphi(v_j)\varphi(x_j) = \prod_{j=1}^{n} u_j \cdot \epsilon = u$. Thus, $v, w \in \varphi^{-1}(L)$. Further, $v \neq \epsilon$ implies that $w \neq \epsilon$.

The fact that $\varphi^{-1}(L)$ is not a T-code now follows, since $w \in \varphi^{-1}(L) \cap (v \sqcup\!\sqcup_t x) \subseteq \varphi^{-1}(L) \cap (\varphi^{-1}(L) \sqcup\!\sqcup_T \Delta^+)$.

For the reverse implication, let $L \subseteq \Sigma^+$ be a T-code such that $\varphi^{-1}(L)$ is not a T-code. Then there exist $t \in T$, $u, v \in \varphi^{-1}(L)$ and $x \in \Delta^+$ such that $v \in u \sqcup\!\sqcup_T x$. As $\varphi(u), \varphi(v) \in L \subseteq \Sigma^+$, $u, v \in \Delta^+$.

Consider $t = \prod_{j=1}^{n} 0^{i_j} 1^{k_j}$ for some $i_j \in I_j$ and $k_j \in K_j$ for $1 \leq j \leq n$. Then $v = \prod_{j=1}^{n} u_j x_j$. for $|u_j| = i_j$, $|x_j| = k_j$, $1 \leq j \leq n$. Consider that

$$\varphi(v) = \prod_{i=1}^{n} \varphi(u_j)\varphi(x_j),$$
$$\varphi(u) = \prod_{i=1}^{n} \varphi(u_j),$$
$$\varphi(x) = \prod_{i=1}^{n} \varphi(x_j).$$

Let $\ell_j = |\varphi(u_j)|$ and $m_j = |\varphi(x_j)|$ for $1 \leq j \leq n$. By assumptions (c) and (d), $\ell_j \in I_j$ and $m_j \in K_j$. Thus,

$$t' = \prod_{j=1}^{n} 0^{\ell_j} 1^{m_j} \in T.$$

Then we may easily observe that

$$\varphi(v) \in \varphi(u) \sqcup_{t'} \varphi(x).$$

As $\varphi(v), \varphi(u) \in L$, a T-code, $\varphi(x) = \epsilon$, and, in particular, $\varphi(x_j) = \epsilon$ for all $1 \leq j \leq n$. Thus, recalling that $k_j = |x_j|$ and $x \neq \epsilon$, we note that

$$(k_1, \cdots, k_n) \in \{|x| \ : \ x \in \varphi^{-1}(\epsilon)\}^n \cap \left(\prod_{j=1}^{n} K_j - \{0\}^n \right).$$

This completes the proof. □

6.3 Closure under Reversal

For a word $w = w_1 w_2 \cdots w_n$, where $w_i \in \Sigma$, its *reversal*, denoted w^R, is given by $w^R = w_n w_{n-1} \cdots w_1$. If $L \subseteq \Sigma^*$ is a language, then its reversal is $L^R = \{w^R \ : \ w \in L\}$. For a class of languages $\mathcal{C}$, let $\mathcal{C}^R = \{L^R \ : \ L \in \mathcal{C}\}$.

Lemma 14. *For all $T \subseteq \{0,1\}^*$, the following equality holds: $\mathcal{P}_{T^R} = \mathcal{P}_T^R$.*

Proof. It suffices to show that $\mathcal{P}_{T^R} \subseteq \mathcal{P}_T^R$.

Let $L \in \mathcal{P}_{T^R}$. Then we have that $L \cap (L \sqcup\!\sqcup_{T^R} \Sigma^+) = \emptyset$. Assume that $L \notin \mathcal{P}_T^R$ and thus $L^R \notin \mathcal{P}_T$. Let $x, y \in L^R$, $t \in T$ and $z \in \Sigma^+$ be such that $x \in y \sqcup\!\sqcup_t z$. Then we note (see, e.g. Mateescu *et al.* [35, Rem. 4.9(ii)]) that $x^R \in y^R \sqcup\!\sqcup_{t^R} z^R$. But as $x^R, y^R \in L$, $t^R \in T^R$, and $z^R \in \Sigma^+$, this contradicts that L is a T^R-code. Thus, $L \in \mathcal{P}_T^R$. □

Corollary 8. *Let $T \subseteq \{0,1\}^*$. Then $\mathcal{P}_T^R = \mathcal{P}_T$ iff $T = T^R$.*

7 Maximal T-codes

Let $T \subseteq \{0,1\}^*$. We say that $L \in \mathcal{P}_T(\Sigma)$ is a *maximal* T-code if, for all $L' \in \mathcal{P}_T(\Sigma)$, $L \subseteq L'$ implies $L = L'$. Denote the set of all maximal T-codes over an alphabet Σ by $\mathcal{M}_T(\Sigma)$. Note that the alphabet Σ is crucial in the definition of maximality. By Zorn's Lemma, we can easily establish that every $L \in \mathcal{P}_T(\Sigma)$ is contained in some element of $\mathcal{M}_T(\Sigma)$.

Again, the proof is a specific instance of a result from dependency theory. Dependency theory is also able to prove the following result; the result is also clear in our case:

Lemma 15. *Let $T_1 \subseteq T_2$. Then for all Σ, $\mathcal{M}_{T_2}(\Sigma) \subseteq \mathcal{M}_{T_1}(\Sigma)$.*

7.1 Decidability and Maximal T-Codes

Unlike showing that every T-code can be embedded in a maximal T-code, to our knowledge, dependency theory has not addressed the problem of deciding whether a language is a maximal code under some dependence system. We address this problem for T-codes now. We first require the following technical lemma, which is interesting in its own right (specific cases were known for, e.g., prefix codes [1, Prop. 3.1, Thm. 3.3] and hypercodes [38, Cor. to Prop. 11]). Let $\tau : \{0,1\}^* \to \{i,d\}^*$ be again given by $\tau(0) = i$ and $\tau(1) = d$.

Lemma 16. *Let $T \subseteq \{0,1\}^*$. Let Σ be an alphabet. For all $L \in \mathcal{P}_T(\Sigma)$, $L \in \mathcal{M}_T(\Sigma)$ iff*

$$L \cup (L \sqcup\!\sqcup_T \Sigma^+) \cup (L \rightsquigarrow_{\tau(T)} \Sigma^+) = \Sigma^+. \tag{7}$$

Proof. Let $L \in \mathcal{P}_T(\Sigma) - \mathcal{M}_T(\Sigma)$. Then there exists $x \in \Sigma^+$ such that $L \cup \{x\} \in \mathcal{P}_T(\Sigma)$, but $x \notin L$. Thus, assume, contrary to what we want to prove, that $x \in (L \sqcup\!\sqcup_T \Sigma^+) \cup (L \rightsquigarrow_{\tau(T)} \Sigma^+)$.

If $x \in L \sqcup\!\sqcup_T \Sigma^+$, then certainly $x \in (L \cup \{x\}) \sqcup\!\sqcup_T \Sigma^+$, by the monotonicity of $\sqcup\!\sqcup_T$. But this contradicts that $L \cup \{x\}$ is a T-code.

If $x \in L \rightsquigarrow_{\tau(T)} \Sigma^+$, then by the monotonicity of $\rightsquigarrow_{\tau(T)}$, $x \in (L \cup \{x\}) \rightsquigarrow_{\tau(T)} \Sigma^+$. But this contradicts that $L \cup \{x\}$ is a T-code, by (1). Thus, $x \notin L \cup (L \sqcup\!\sqcup_T \Sigma^+) \cup (L \rightsquigarrow_{\tau(T)} \Sigma^+)$.

For the reverse implication, assume that $L \in \mathcal{M}_T(\Sigma)$. Then for all $x \in \Sigma^+$ with $x \notin L$, there exist $y \in L, z \in \Sigma^+$ such that either $x \in y \sqcup\!\sqcup_T z$ or $y \in x \sqcup\!\sqcup_T z$. The second membership is equivalent to $x \in y \rightsquigarrow_{\tau(T)} z$. Thus, we have $x \in (L \sqcup\!\sqcup_T \Sigma^+) \cup (L \rightsquigarrow_{\tau(T)} \Sigma^+)$ for all $x \in \Sigma^+ - L$. The result then follows. □

Corollary 9. *Let $T \subseteq \{0,1\}^*$ be a regular set of trajectories. Given a regular language $L \subseteq \Sigma^+$, it is decidable whether $L \in \mathcal{M}_T(\Sigma)$.*

Proof. By Lemma 3, we can decide whether $L \in \mathcal{P}_T(\Sigma)$. If not, then certainly $L \notin \mathcal{M}_T(\Sigma)$. Otherwise, since T, L are regular, then the languages $L, L \sqcup\!\sqcup_T \Sigma^+$, $L \rightsquigarrow_{\tau(T)} \Sigma^+$ and $L \cup (L \sqcup\!\sqcup_T \Sigma^+) \cup (L \rightsquigarrow_{\tau(T)} \Sigma^+)$ are regular. Thus, the equality (7) is decidable. □

Similar results were also obtained by Kari *et al.* [27, Sect. 5].

We now consider the decidability of being a maximal T-code for finite languages. Our goal is to give a class of sets of trajectories larger than REG such that for any T in our class, it is decidable whether an arbitrary finite language is a maximal T-code.

We first introduce some notation. Let $T \subseteq \{0,1\}^*$. For any $n \geq 0$, let $\eta_n(T) = \{t \in T \ : \ |t|_0 = n\}$. Clearly, $\cup_{n \geq 0} \eta_n(T) = T$.

Before we begin, we require some preliminary lemmas. Recall that a semilinear set over $\mathbb{N}^k$ is a finite union of sets of the form $\{\mathbf{u} + \sum_{i=1}^n c_i \mathbf{v_i} \ : \ c_i \in \mathbb{N}\}$ where $\mathbf{u}, \mathbf{v_i} \in \mathbb{N}^k$. The following lemma can be found in Ginsburg [10, Cor. 5.3.2]:

Lemma 17. *Let $T \subseteq w_1^* w_2^*$ for $w_1, w_2 \in \{0,1\}^*$. Then T is a CFL iff $\{(m,n) \ : \ w_1^m w_2^n \in T\}$ is a semilinear set.*

Lemma 18. *Let $T \subseteq w_1^* w_2^*$ for $w_1, w_2 \in \{0,1\}^*$. If w_1, w_2 are given and T is an effectively given CFL, then for all $n \geq 1$, $\eta_n(T)$ is an effectively regular language.*

For example, let $T = \{0^m 1^m \ : \ m \geq 0\} \subseteq 0^*1^*$. Then $\eta_n(T) = \{0^n 1^n\}$ for all $n \geq 0$. If $T = (01)^*1^*$, then $\eta_n(T) = (01)^n 1^*$. We note that we cannot relax the conditions of Lemma 18 to $T \subseteq w_1^* w_2^* w_3^*$, since, e.g., $T = \{1^n 0^m 1^n \ : \ n, m \geq 0\} \subseteq 1^*0^*1^*$, but $\eta_m(T) = \{1^n 0^m 1^n \ : \ n \geq 0\}$, which is not regular.

Proof. Let $T \subseteq w_1^* w_2^*$ for $w_1, w_2 \in \{0,1\}^*$. Let S be the semilinear set such that $w_1^{\alpha_1} w_2^{\alpha_2} \in T$ iff $(\alpha_1, \alpha_2) \in S$. Since the union of regular languages is regular, we can assume without loss of generality that S

is linear, i.e., there exist $m, k_1, k_2 \geq 0$ and $p_1, r_i \geq 0$ for all $1 \leq i \leq m$ such that

$$S = \{(k_1, k_2) + \sum_{i=1}^{m} n_i(p_i, r_i) \ : \ (n_1, \ldots, n_m) \in \mathbb{N}^m\}.$$

We assume without loss of generality that $(p_j, r_j) \neq (0, 0)$ for all $1 \leq j \leq m$, otherwise, we can simply remove this index from our set without affecting S. We distinguish between four cases:

(a) $w_1 w_2 \in 1^* + 0^*$. In this case, as T is a unary CFL, it is known that T is a regular language. Thus, so is $\eta_n(T) = T \cap (1^*0)^n 1^*$.

(b) $w_1 \in 1^*$. By case (a), we can assume that $w_2 \notin 1^*$, i.e., that $|w_2|_0 \neq 0$.
As $w_1 \in 1^*$, there exists $\alpha \geq 0$ such that

$$T = \{1^{\alpha(k_1 + \sum_{i=1}^{m} n_i p_i)} w_2^{k_2 + \sum_{i=1}^{m} n_i r_i} \ : \ (n_1, \ldots, n_m) \in \mathbb{N}^m\}.$$

Let $I \subseteq \mathbb{N}^m$ be defined so that

$$I = \{(n_1, \ldots, n_m) \ : \ |w_2|_0 (k_2 + \sum_{i=1}^{m} n_i r_i) = n\}.$$

From this, we can see that

$$\eta_n(T) = \{1^{\alpha(k_1 + \sum_{i=1}^{m} n_i p_i)} w_2^{k_2 + \sum_{i=1}^{m} n_i r_i} \ : \ (n_1, \ldots, n_m) \in I\}.$$

By reordering if necessary, let $0 \leq m' \leq m$ be the index such that for all $j \leq m'$, $r_j \neq 0$ and for all $m' < j \leq m$, $r_j = 0$. Let $\varphi : I \to \mathbb{N}^{m'}$ be given by $\varphi(n_1, n_2, \ldots, n_m) = (n_1, n_2, \ldots, n_{m'})$. Note that $\varphi^{-1}(\varphi(I)) = I$ as we have that if $(n_1, \ldots, n_m) \in I$, for all $m' < j \leq m$,

$$(n_1, n_2, \ldots, n_{j-1}, n'_j, n_{j+1}, \ldots, n_m) \in I$$

for all $n'_j \in \mathbb{N}$.
Further, note that $\varphi(I)$ is finite, since for all $(n_1, \ldots, n_{m'}) \in \varphi(I)$ and all $j \leq m'$, n_j satisfies

$$n_j \leq \frac{1}{r_j} \left(\frac{n}{|w_2|_0} - k_2 \right).$$

Thus, we can conclude that

$$\begin{aligned} \eta_n(T) = \{1^{\alpha(k_1 + \sum_{i=1}^{m'} n_i p_i)} (\prod_{i=m'+1}^{m} (1^{\alpha p_i})^*) w_2^{k_2 + \sum_{i=1}^{m'} n_i r_i} \\ : \ (n_1, \ldots, n_{m'}) \in \varphi(I)\}. \end{aligned}$$

and that $\eta_n(T)$ is regular.

(c) $w_2 \in 1^*$. By (a), $w_1 \notin 1^*$. Thus, consider that $\eta_n(T^R) = \eta_n(T)^R$. As $T^R \subseteq (w_2^R)^*(w_1^R)^*$, by (b), $\eta_n(T^R)$ is regular. As the regular languages are closed under reversal, $\eta_n(T)$ is regular.

(d) $w_1, w_2 \notin 1^*$. Let $I \subseteq \mathbb{N}^m$ be defined by

$$I = \{(n_1, \ldots, n_m) \in \mathbb{N}^m : |w_1|_0(k_1 + \sum_{i=1}^{m} n_i p_i) + |w_2|_0(k_2 + \sum_{i=1}^{m} n_i r_i) = n\}.$$

Note that I is finite, as $|w_1|_0, |w_2|_0 \neq 0$ and $(p_i, r_i) \neq (0,0)$ for all $1 \leq i \leq m$. Further, we have that

$$\eta_n(T) = \{w_1^{k_1 + \sum_{i=1}^m n_i p_i} w_2^{k_2 + \sum_{i=1}^m n_i r_i} : (n_1, \ldots, n_m) \in I\}.$$

From this, we note that $\eta_n(T)$ is finite.

Thus, $\eta_n(T)$ is regular. □

We are now ready to give our positive decidability result:

Theorem 7. *Let $T \subseteq \{0,1\}^*$ be an effectively given CFL such that $T \subseteq w_1^* w_2^*$ for $w_1, w_2 \in \{0,1\}^*$, where w_1, w_2 are given. If F is an effectively given finite set, then we can decide whether F is a maximal T-code.*

Proof. Let $T \subseteq w_1^* w_2^*$ be a CFL. Let F be our finite set and let $\ell(F) = \{|x| : x \in F\}$ and $\ell_F = \max\{\ell : \ell \in \ell(F)\}$. First, we note that we can effectively find $T^{\leq \ell_F} = T \cap \{0,1\}^{\leq \ell_F}$, and that

$$F \rightsquigarrow_{\tau(T)} \Sigma^+ = F \rightsquigarrow_{\tau(T^{\leq \ell_F})} \Sigma^+,$$

which is thus an effectively regular language, since $F, \Sigma^+, \tau(T^{\leq \ell_F})$ are, as well.

Second, we note that $\eta(T) = \cup_{\ell \in \ell(F)} \eta_\ell(T)$ is an effectively regular language, since $\ell(F)$ is effectively finite, and $\eta_\ell(T)$ is effectively regular by Lemma 18. Further, we note that

$$F \sqcup\!\sqcup_T \Sigma^+ = F \sqcup\!\sqcup_{\eta(T)} \Sigma^+,$$

which is a regular language, by the regularity of F, Σ^+ and $\eta(T)$.

Thus, we conclude that $F \cup (F \rightsquigarrow_{\tau(T)} \Sigma^+) \cup (F \sqcup\!\sqcup_T \Sigma^+)$ is an effectively regular language, and thus, we can determine whether this language is equal to Σ^+. Thus, by Lemma 16, we can determine whether F is a maximal T-code. □

7.2 Transitivity and Embedding T-codes

Given a class of codes $\mathcal{C}$, and a language $L \in \mathcal{C}$ of given complexity, there has been much research into whether or not L can be *embedded in* (or *completed to*) a maximal element $L' \in \mathcal{C}$ of the same complexity, i.e., a maximal code L' with $L \subseteq L'$. Finite and regular languages of these classes of codes are of particular interest. For instance, we note that every regular code can be completed to a maximal regular code, while the same is not true for finite codes or finite biprefix codes.

We now show an interesting result on embedding T-codes in maximal T-codes, while preserving complexity. For example, we will show that if T is transitive and regular and L is a regular T-code, then we can embed L in a maximal T-code which is also regular.

Our construction is a generalization of a result due to Lam [31]. In particular, we define two transformations on languages. Let T be a set of trajectories and $L \subseteq \Sigma^+$ be a language. Then define $U_T(L), V_T(L) \subseteq \Sigma^+$ as

$$U_T(L) = \Sigma^+ - (L \sqcup\!\sqcup_T \Sigma^+ \cup L \rightsquigarrow_{\tau(T)} \Sigma^+);$$
$$V_T(L) = U_T(L) - (U_T(L) \sqcup\!\sqcup_T \Sigma^+).$$

First, we note the following two properties of $U_T(L), V_T(L)$:

Lemma 19. *Let $T \subseteq \{0,1\}^*$ be a set of trajectories and $L \in \mathcal{P}_T(\Sigma)$. Then $L \subseteq U_T(L)$ and $L \subseteq V_T(L)$.*

Proof. We establish first that $L \subseteq U_T(L)$. Let $x \in L$, but assume that $x \notin U_T(L)$. Then $x \in L \sqcup\!\sqcup_T \Sigma^+$ or $x \in L \rightsquigarrow_{\tau(T)} \Sigma^+$. In the first case, we have $L \cap (L \sqcup\!\sqcup_T \Sigma^+) \neq \emptyset$, contradicting that L is a T-code. The second case also contradicts that L is a T-code, since then $L \cap (L \rightsquigarrow_{\tau(T)} \Sigma^+) \neq \emptyset$, contradicting (1).

We now establish $L \subseteq V_T(L)$. Assume not, then as $L \subseteq U_T(L)$, we must have that $L \cap (U_T(L) \sqcup\!\sqcup_T \Sigma^+) \neq \emptyset$. Assume that $y \in U_T(L)$, $z \in \Sigma^+$ and $x \in L$ are chosen so that $x \in y \sqcup\!\sqcup_T z$. Therefore, we have that $y \in x \rightsquigarrow_{\tau(T)} z \subseteq L \rightsquigarrow_{\tau(T)} \Sigma^+$, contradicting that $y \in U_T(L)$. Thus, $L \subseteq V_T(L)$. □

Theorem 8. *Let $T \subseteq \{0,1\}^*$ be transitive. Let Σ be an alphabet. Then for all $L \in \mathcal{P}_T(\Sigma)$, the language $V_T(L)$ contains L and $V_T(L) \in \mathcal{M}_T(\Sigma)$.*

Proof. By Lemma 19, $L \subseteq V_T(L)$. That $V_T(L)$ is a T-code follows from Lemma 5 applied to $U_T(L)$. Thus, it remains to show that for all $z \in \Sigma^+$ with $z \notin V_T(L)$, $V_T(L) \cup \{z\}$ is not a T-code.

Let $z \notin V_T(L)$ be arbitrary. We distinguish two cases:

(a) if $z \notin U_T(L)$, then $z \in (L \sqcup\!\sqcup_T \Sigma^+) \cup (L \rightsquigarrow_{\tau(T)} \Sigma^+)$. If $z \in L \sqcup\!\sqcup_T \Sigma^+ \subseteq V_T(L) \sqcup\!\sqcup_T \Sigma^+$, then $V_T(L) \cup \{z\} \notin \mathcal{P}_T(\Sigma)$. If $z \in L \rightsquigarrow_{\tau(T)} \Sigma^+ \subseteq V_T(L) \rightsquigarrow_{\tau(T)} \Sigma^+$, then again (this time by (1)), $V_T(L) \cup \{z\} \notin \mathcal{P}_T(\Sigma)$.

(b) if $z \in U_T(L) - V_T(L)$, then $z \in U_T(L) \sqcup\!\sqcup_T \Sigma^+$. Let $y \in U_T(L)$ be a shortest word such that $z \in y \sqcup\!\sqcup_T \Sigma^+$. We claim that $y \in V_T(L)$. If this were not the case, then as $y \in U_T(L) - V_T(L)$, we have that $y \in U_T(L) \sqcup\!\sqcup_T \Sigma^+$, by definition of $V_T(L)$. Let $y' \in U_T(L)$ be such that $y \in y' \sqcup\!\sqcup_T \Sigma^+$. Thus, we have that $y' \omega_T y \omega_T z$. By transitivity of T, $y' \omega_T z$, i.e., $z \in y' \sqcup\!\sqcup_T \Sigma^*$. As $|y'| < |y| < |z|$, we certainly have that $z \in y' \sqcup\!\sqcup_T \Sigma^+$ in particular. But as $|y'| < |y|$, this contradicts our choice of y. Thus, $y \in V_T(L)$. But $y, z \in V_T(L) \cup \{z\}$ and $z \in y \sqcup\!\sqcup_T \Sigma^+$ imply that $V_T(L) \cup \{z\} \notin \mathcal{P}_T(\Sigma)$.

Thus, $V_T(L)$ is a maximal T-code. □

There are several important consequences of Theorem 8. We note only one important corollary:

Corollary 10. *Let $T \subseteq \{0,1\}^*$ be transitive and regular. Then every regular (resp., recursive) T-code is contained in a maximal regular (resp., recursive) T-code.*

Corollary 10 was given for $T = 1^*0^*1^*$ and regular T-codes by Lam [31, Prop. 3.2]. Further research into the case when T is not transitive is necessary (for example, the proofs of Zhang and Shen [45] and Bruyère and Perrin [2] on embedding regular biprefix codes are much more involved than our construction, and do not seem to be easily generalized).

We can extend our embedding results to finite languages with one additional constraint on T. Recall that T is said to be complete if for all $n_1, n_2 \in \mathbb{N}$, there exists $t \in T$ such that $|t|_0 = n_1$ and $|t|_1 = n_2$. The following technical lemma is easily proven:

Lemma 20. *Let $T \subseteq \{0,1\}^*$ be complete. Then for all $y \in \Sigma^*$ and for all $m \leq |y|$, there exists $z \in \Sigma^m$ such that $y \in z \sqcup\!\sqcup_T \Sigma^*$. Further, if $m < |y|$, $y \in z \sqcup\!\sqcup_T \Sigma^+$.*

We now show that for transitive and complete sets of trajectories T, finite T-codes can be completed to finite maximal T-codes.

Corollary 11. *Let $T \subseteq \{0,1\}^*$ be transitive and complete. Let Σ be an alphabet. Then for all finite $F \in \mathcal{P}_T(\Sigma)$, there exists a finite language $F' \in \mathcal{M}_T(\Sigma)$ such that $F \subseteq F'$. Further, if T is effectively regular, and F is effectively given, we can effectively construct F'.*

Proof. Let F be a finite language and $n = \max\{|x| : x \in F\}$. As $F \in \mathcal{P}_T(\Sigma)$, $n \neq 0$. We first establish the following claim: for all $y \in \Sigma^+$ with $|y| > n$, there exists $u \in U_T(F)$ such that $y \in u \sqcup\!\sqcup_T \Sigma^+$.

Let $y \in \Sigma^+$ be such that $|y| > n$. Then by Lemma 20, there exists z such that $|z| = n$ and $y \in z \sqcup\!\sqcup_T \Sigma^+$. Note that as $n \neq 0$, $z \in \Sigma^+$. If $z \in U_T(F)$, we have established the claim with $u = z$. Thus, assume that $z \notin U_T(F)$. By definition of $U_T(F)$, we have that $z \in (F \sqcup\!\sqcup_T \Sigma^+) \cup (F \rightsquigarrow_{\tau(T)} \Sigma^+)$. However, $|x| < n$ holds for all $x \in F \rightsquigarrow_{\tau(T)} \Sigma^+$. Thus, we have that $z \in F \sqcup\!\sqcup_T \Sigma^+ \subseteq U_T(F) \sqcup\!\sqcup_T \Sigma^+$, the inclusion being valid by Lemma 19. Let $u \in U_T(F)$ be such that $z \in u \sqcup\!\sqcup_T \Sigma^+$. Then $u \omega_T z$ and $z \omega_T y$. Thus, by transitivity, $u \omega_T y$. As $|u| < |y|$, this implies that $y \in u \sqcup\!\sqcup_T \Sigma^+$. Thus, our claim is proven.

We now establish that $V_T(F)$ is finite. Let y be an arbitrary word such that $|y| > n$. By our claim, $y \in U_T(F) \sqcup\!\sqcup_T \Sigma^+$. But by definition of $V_T(F)$, this implies that $y \notin V_T(F)$. Thus, $V_T(F) \subseteq \Sigma^{\leq n}$. Thus, the conditions of the corollary are met by $V_T(F)$. This completes the proof. □

In practice, the condition that T be complete is not very restrictive, since natural operations seem to typically be defined by a complete set of trajectories.

In Section 8.3 below, we will give alternate conditions on T that ensure that every regular T-code can be embedded in a finite maximal T-code. However, this result will be a trivial consequence of the fact that for such T, all T-codes are finite.

We now show the existence of T which are not transitive, and for which the above results do not hold. It is known, for example, that there exist finite biprefix codes which cannot be embedded in a maximal finite biprefix code (see, e.g., Bruyère and Perrin [2, Sect. 3]). We present the following two examples, as well; in the first case, T is regular but not transitive, and for all finite T-codes L, L cannot be embedded in any maximal CF T-code. In the second example, T is not complete, and no finite T-code can be embedded in a maximal finite T-code.

Example 2. Let $T = (01)^*$; then $\sqcup\!\sqcup_T$ is known as perfect or balanced literal shuffle. Clearly, T is not transitive. Let $\Sigma = \{a\}$. We claim that for all regular languages $L \subseteq a^*$, L is not a maximal T-code.

Let $L \subseteq a^*$ be regular. As L is a unary regular language, it is well-known that L corresponds to an ultimately periodic set of natural numbers. That is, there exist $n_0, p \in \mathbb{N}$ with $p > 0$ such that for all $n > n_0$, $a^n \in L$ iff $a^{n+p} \in L$.

Let $r = \min\{kp : k \geq 1, kp > n_0\}$. Then we have two cases:

(a) if $a^r \in L$, then $a^{2r} \in L$ as well. Thus, as $a^{2r} \in a^r \sqcup\!\sqcup_T a^r$, L is not a T-code.
(b) if $a^r \notin L$, as $r > n_0$, $a^{2r} \notin L$ as well. Thus, consider $L \cup \{a^{2r}\}$. If L is a T-code, then as $a^{2r} \notin L \sqcup\!\sqcup_T a^+$ and $L \cap (a^{2r} \sqcup\!\sqcup_T a^+) = \emptyset$, we have that $L \cup \{a^{2r}\}$ is a T-code as well. Thus, L is not a maximal T-code.

Thus, there are no regular languages in $\mathcal{M}_T(\{a\})$ (and hence, no context-free languages in $\mathcal{M}_T(\{a\})$, since the unary context-free and unary regular languages coincide). Thus, e.g., the T-code $\{a\}$ cannot be embedded in any regular (or context-free) maximal T-code.

We note in passing that one maximal T-code containing $\{a\}$ is given by $L = \{a^{c_n} \ : \ n \geq 1\}$ where $\{c_n\}_{n\geq 1} = \{1, 3, 4, 5, 7, 9, 11, \dots\}$ is the lexicographically least sequence of positive integers satisfying $m \in \{c_n\} \iff 2m \notin \{c_n\}$. This sequence has received some attention in the literature, and has connections to the Thue-Morse word. We point the reader to A003159 in Sloane [40] for details and references. Clearly, L is not regular.

Example 3. Let $T = \{0^j 1^{2i} 0^j \ : \ i, j \geq 0\}$. Then $\sqcup\!\sqcup_T$ is the balanced insertion operation. Note that T is transitive, but not complete. Let Σ be an alphabet and let $L_o = \{x \in \Sigma^+ \ : \ |x| \equiv 1 \pmod 2)\}$. Then for all $L \in \mathcal{P}_T(\Sigma)$, $L \cup L_o \in \mathcal{P}_T(\Sigma)$. Thus, there are no finite maximal T-codes.

8 Finiteness of T-codes

In this section, we investigate $T \subseteq \{0,1\}^*$ such that all $\mathcal{P}_T$ codes are finite. It is a well-known result that all hypercodes ($T = \{0,1\}^*$) are finite, which can be concluded from a result due to Higman [15].

We define the following classes of sets of trajectories:

$$\mathfrak{F}_R = \{T \in \{0,1\}^* \ : \ \mathcal{P}_T \cap \text{REG} \subseteq \text{FIN}\};$$
$$\mathfrak{F}_C = \{T \in \{0,1\}^* \ : \ \mathcal{P}_T \cap \text{CF} \subseteq \text{FIN}\};$$
$$\mathfrak{F}_H = \{T \in \{0,1\}^* \ : \ \mathcal{P}_T \subseteq \text{FIN}\}.$$

The class $\mathfrak{F}_H$ is of particular importance. If T is a partial order and $T \in \mathfrak{F}_H$, then T is a *well partial order*[1]. This is a subject of tremendous research, not only in the larger theory of partial orders (see the survey of Kruskal [30]), but also within formal language theory as well. Without trying to be exhaustive, we note the work of Jullien [21], Haines [12], van Leeuwen [43], Ehrenfeucht *et al.* [8], Ilie

[1] Recall that we say that T has property P iff ω_T has property P.

[17,18], Ilie and Salomaa [19] and Harju and Ilie [13] on well partial orders relating to words. We also refer the reader to the survey of results presented by de Luca and Varricchio [5, Sect. 5].

To begin, we give conditions on T which ensure all regular (or context-free) T-codes are finite.

8.1 Finiteness of Regular T-codes

Let $T \subseteq \{0,1\}^*$. Define the *insertion behaviour* of T, denoted $ib(T)$, as

$$ib(T) = \{(n_1, n_2, n_3) \in \mathbb{N}^3 \ : \ 0^{n_1}1^{n_2}0^{n_3} \in T\}.$$

Say that T is REG-*pumping compliant* if, for all $i, j, k \in \mathbb{N}$ $(j > 0)$, there exists j' with $0 \leq j' < j$ such that

(i) if $j' = 0$, then $ib(T) \cap \{(i + jm_1, jm_2, k + jm_3) \ : \ m_1, m_3 \geq 0, m_2 > 0\} \neq \emptyset$.

(ii) if $1 \leq j' < j$, then $ib(T) \cap \{(i + j' + jm_1, jm_2, k - j' + jm_3) \ : \ m_1 \geq 0, m_2, m_3 > 0\} \neq \emptyset$.

Lemma 21. *Let $T \subseteq \{0,1\}^*$. If T is* REG*-pumping compliant, then $T \in \mathfrak{F}_R$.*

Proof. Let $R \in$ REG be an infinite regular language over Σ. By the pumping lemma for regular languages, there exist $u, v, w \in \Sigma^*$ such that $v \neq \epsilon$ and $uv^*w \subseteq R$. Let $i = |u|$, $j = |v|$ and $k = |w|$. Note that $j \neq 0$. Let j' be the natural number implied by the REG-pumping compliance condition.

If $j' = 0$, then let m_1, m_2, m_3 be chosen so that $m_1, m_3 \geq 0$, $m_2 > 0$ and $(i + jm_1, jm_2, k + jm_3) \in ib(T)$. Let $t = 0^{i+jm_1}1^{jm_2}0^{k+jm_3}$. By definition, $t \in T$. Consider $x = uv^{m_1+m_3}w \in R$ and $y = v^{m_2}$. As $m_2 \neq 0$ and $v \neq \epsilon$, $y \neq \epsilon$. We note that

$$x \sqcup\!\sqcup_t y \ni uv^{m_1} \cdot v^{m_2} \cdot v^{m_3}w = uv^{m_1+m_2+m_3}w.$$

Thus, $(R \sqcup\!\sqcup_T \Sigma^+) \cap R \neq \emptyset$ and $R \notin \mathcal{P}_T$.

If $1 \leq j' < j$, let $m_1 \geq 0$, $m_2, m_3 > 0$ be chosen so that

$$(i + j' + jm_1, jm_2, k - j' + jm_3) \in ib(T),$$

and hence $t = 0^{i+j'+jm_1}1^{jm_2}0^{k+(j-j')+j(m_3-1)} \in T$. Let $v_1 \in \Sigma^*$ be the prefix of v of length j' and let $v = v_1v_2$ for some $v_2 \in \Sigma^*$.

Consider $x = uv^{m_1+m_3}w \in R$ and $y = (v_2v_1)^{m_2} \neq \epsilon$. Then

$$x \sqcup\!\sqcup_t y \ni uv^{m_1}v_1 \cdot v_2(v_1v_2)^{m_2-1}v_1 \cdot v_2v^{m_3-1}w = uv^{m_1+m_2+m_3}w.$$

Again, $(R \sqcup\!\sqcup_T \Sigma^+) \cap R \neq \emptyset$ and thus $R \notin \mathcal{P}_T$. Thus, $\mathcal{P}_T$ contains no infinite regular languages. $\square$

The condition of being REG-pumping compliant is not very restrictive. Clearly, if $T \supseteq 0^*1^*0^*$, then T is REG-pumping compliant (in this case, Lemma 21 is a corollary of a result on outfix codes due to Ito *et al.* [20]). For a broader class of examples, we can consider *immune* languages [9]. Let $\mathcal{C}$ be a class of languages. A language L is *$\mathcal{C}$-immune* if L is infinite and for all infinite subsets $L' \subseteq L$, $L' \notin \mathcal{C}$.

Lemma 22. *Let $T \subseteq \{0,1\}^*$ be a set of trajectories such that $\overline{T} \cap 0^*1^*0^*$ is REG-immune. Then T is REG-pumping compliant.*

Proof. Let $i \geq 0, j > 0, k \geq 0$ be arbitrary. Consider

$$T_0 = T_0(i,j,k) = 0^i(0^j)^*(1^j)^+(0^j)^*0^k.$$

As T_0 is a regular language, T_0 is not a subset of $\overline{T} \cap 0^*1^*0^*$. Thus, $T_0 \cap (\overline{\overline{T} \cap 0^*1^*0^*}) = T_0 \cap (T \cup \overline{0^*1^*0^*}) \neq \emptyset$. As $T_0 \subseteq 0^*1^*0^*$, this implies that $T_0 \cap T \neq \emptyset$. Thus, there exist $m_1 \geq 0$, $m_2 > 0$ and $m_3 \geq 0$ such that $0^{i+jm_1}1^{jm_2}0^{k+jm_3} \in T$, i.e., $(i + jm_1, jm_2, k + jm_3) \in ib(T)$. Thus, the REG-pumping compliant conditions are met with $j' = 0$. □

Next, we show that if $T \subseteq 0^*1^*0^*$, then REG-pumping compliance is necessary to ensure that there are no infinite regular languages in $\mathcal{P}_T$.

Lemma 23. *Let $T \subseteq 0^*1^*0^*$ be not REG-pumping compliant. Then $\mathcal{P}_T(\Sigma)$ contains an infinite regular language for all Σ with $|\Sigma| > 1$.*

Proof. Let $i, j, k \in \mathbb{N}$ be arbitrary such that $i \geq 0, j > 0, k \geq 0$,

$$ib(T) \cap \{(i + jm_1, jm_2, k + jm_3) \ : \ m_1, m_3 \geq 0, m_2 > 0\} = \emptyset.$$

and for all $1 \leq j' < j$,

$$ib(T) \cap \{(i + j' + jm_1, jm_2, k - j' + jm_3) \ : \ m_1 \geq 0, m_2, m_3 > 0\} = \emptyset.$$

Let $a, b \in \Sigma$ be distinct letters and $R = a^i(b^j)^*a^k$. We claim that $R \in \mathcal{P}_T(\Sigma)$. Assume not. Then there exist $\ell_1 > \ell_2 \geq 0$ such that

$$a^ib^{j\ell_1}a^k \in a^ib^{j\ell_2}a^k \sqcup\!\sqcup_T z$$

for some $z \in \{a,b\}^+$. By observation, $z = b^{j(\ell_1 - \ell_2)}$. Thus, let $t \in T$ be chosen so that

$$a^ib^{j\ell_1}a^k \in a^ib^{j\ell_2}a^k \sqcup\!\sqcup_t b^{j(\ell_1-\ell_2)}.$$

Then as $T \subseteq 0^*1^*0^*$, $t = 0^{i+\alpha j+j'}1^{j(\ell_1-\ell_2)}0^{(j-j')+(\ell_2-\alpha-1)j+k}$ for some α and j' with either $0 \leq \alpha \leq \ell_2$ and $j' = 0$ or $0 \leq \alpha < \ell_2 - 1$ and $1 \leq j' < j$. If $j' = 0$, then $(i + \alpha j, j(\ell_1 - \ell_2), k + (\ell_2 - \alpha)j) \in ib(T)$ while if $j' \neq 0$, then $(i + j' + \alpha j, j(\ell_1 - \ell_2), k - j' + (\ell_2 - \alpha)j) \in ib(T)$, which are both contradictions. □

8.2 Finiteness of Context-free T-codes

Let $T \subseteq \{0,1\}^*$. Define the *2–insertion behaviour* of T, denoted $2ib(T)$, as follows:

$$2ib(T) = \{(n_1, n_2, \ldots, n_5) \in \mathbb{N}^5 \ : \ 0^{n_1}1^{n_2}0^{n_3}1^{n_4}0^{n_5} \in T\}.$$

We use $2ib(T)$ to define the notion of CF-*pumping compliance*. The idea is the same as REG-pumping compliance, but with more cases. In particular, say that T is CF-pumping compliant if, for all $i, j_1, j_2, k, \ell \in \mathbb{N}$, with $j_1 + j_2 > 0$, there exist $j_1', j_2' \in \mathbb{N}$ such that $0 \leq j_i' < j_i$ for $i = 1, 2$ and $2ib(T) \cap P \neq \emptyset$, where P is defined as follows:

(a) if $j_1' = j_2' = 0$, then

$$P = \{(i + j_1\alpha_1, j_1\beta, k + j_1\alpha_2 + j_2\alpha_3, j_2\beta, \ell + j_2\alpha_4) \\ : \ \alpha_m, \beta \in \mathbb{N}, (1 \leq m \leq 4), \beta > 0, \alpha_1 + \alpha_2 = \alpha_3 + \alpha_4\}.$$

(b) if $1 \leq j_1' < j_1$ and $j_2' = 0$, then P is defined by the set

$$\{(i + j_1' + j_1\alpha_1, j_1\beta, k - j_1' + j_1\alpha_2 + j_2(\alpha_3 + \gamma_1), j_2\beta, \ell + j_2(\alpha_4 + \gamma_2)) \\ : \ \alpha_m, \beta, \gamma_p \in \mathbb{N}, (1 \leq m \leq 4, 1 \leq p \leq 2), \\ \beta, \alpha_2 > 0, \alpha_1 + \alpha_2 = \alpha_3 + \alpha_4 + 1, \gamma_1 + \gamma_2 = 1\}.$$

(c) if $j_1' = 0$ and $1 \leq j_2' < j_2$, then P is defined by the set

$$\{(i + j_1(\alpha_1 + \gamma_1), j_1\beta, k + j_2' + j_1(\alpha_2 + \gamma_2) + j_2\alpha_3, j_2\beta, \ell - j_2' + j_2\alpha_4) \\ : \ \alpha_m, \beta, \gamma_p \in \mathbb{N}, (1 \leq m \leq 4, 1 \leq p \leq 2), \\ \beta, \alpha_4 > 0, \alpha_1 + \alpha_2 + 1 = \alpha_3 + \alpha_4, \gamma_1 + \gamma_2 = 1\}.$$

(d) if $1 \leq j_1' < j_1$ and $1 \leq j_2' < j_2$, then P is defined by the set

$$\{(i + j_1' + j_1\alpha_1, j_1\beta, k - j_1' + j_2' + j_1\alpha_2 + j_2\alpha_3, j_2\beta, \ell - j_2' + j_2\alpha_4) \\ : \ \alpha_m, \beta \in \mathbb{N}, (1 \leq m \leq 4), \ \beta, \alpha_2, \alpha_4 > 0, \alpha_1 + \alpha_2 = \alpha_3 + \alpha_4\}.$$

Lemma 24. *Let $T \subseteq \{0,1\}^*$. If T is* CF*-pumping compliant, then $T \in \mathfrak{F}_C$.*

Proof. Let $L \in$ CF be an infinite language which is a subset of Σ^+. Then by the pumping lemma for CFLs, there exist $u, v, w, x, y \in \Sigma^*$ such that $vx \neq \epsilon$ and $\{uv^m wx^m y \ : \ m \geq 0\} \subseteq L$. Let $i = |u|$, $j_1 = |v|$, $k = |w|$, $j_2 = |x|$ and $\ell = |y|$. Let j_1', j_2' be the natural numbers implied by the CF-pumping compliance of T. We consider the case $j_1' = 0$ and $1 \leq j_2' < j_2$. The other cases are similar (the differences are similar to the differences between the cases in the proof of Lemma 21).

Let $\alpha_m, \beta, \gamma_p \in \mathbb{N}$ for $1 \leq m \leq 4$ and $1 \leq p \leq 2$ be such that $2ib(T)$ contains the element

$$(i + j_1(\alpha_1 + \gamma_1), j_1\beta, k + j_2' + j_1(\alpha_2 + \gamma_2) + j_2\alpha_3, j_2\beta, \ell - j_2' + j_2\alpha_4).$$

Further, we have that $\beta, \alpha_4 > 0$, $\alpha_1 + \alpha_2 + 1 = \alpha_3 + \alpha_4$ and $\gamma_1 + \gamma_2 = 1$, i.e., one of $\gamma_p = 0$ and other is equal to one. Consider that

$$uv^{\alpha_1+\alpha_2+1} wx^{\alpha_3+\alpha_4} y, uv^{\alpha_1+\alpha_2+1+\beta} wx^{\alpha_3+\alpha_4+\beta} y \in L.$$

Further, if $x = x_1 x_2$ where $x_1, x_2 \in \Sigma^*$ and $|x_1| = j_2'$, then

$$uv^{\alpha_1+\alpha_2+1+\beta} wx^{\alpha_3+\alpha_4+\beta} y \in z_1 \cdot z_2 \cdot z_3 \sqcup\!\sqcup_t v^\beta (x_2 x_1)^\beta$$

where

$$\begin{aligned} z_1 &= uv^{\alpha_1+\gamma_1}, \\ z_2 &= v^{\alpha_2+\gamma_2} wx^{\alpha_3} x_1, \\ z_3 &= x_2 x^{\alpha_4 - 1} y, \\ t &= 0^{i+j_1(\alpha_1+\gamma_1)} 1^{j_1\beta} 0^{k+j_2'+j_1(\alpha_2+\gamma_2)+j_2\alpha_3} 1^{j_2\beta} 0^{\ell - j_2' + j_2\alpha_4} \in T. \end{aligned}$$

Note that

$$z_1 z_2 z_3 = uv^{\alpha_1+\alpha_2+1} wx^{\alpha_3+\alpha_4} y \in L.$$

As $vx \neq \epsilon$ and $\beta > 0$, $v^\beta (x_2 x_1)^\beta \neq \epsilon$. Thus, $L \notin \mathcal{P}_T$. $\square$

Note that if $T \supseteq 0^*1^*0^*1^*0^*$ then T satisfies the conditions of Lemma 24. This instance of our result is also a corollary of a result due to Thierrin and Yu [42, Prop. 3.3(2)].

8.3 Finiteness of T-codes

We now turn to the question of the existence of arbitrary infinite languages in a class of T-codes. We first show that if T is bounded, then there is an infinite T-code.

Theorem 9. *Let $T \subseteq \{0,1\}^*$ be a bounded language. Then for all Σ with $|\Sigma| > 1$, $\mathcal{P}_T(\Sigma)$ contains an infinite language.*

Proof. Let $T \subseteq \{0,1\}^*$ be a bounded language. Then there exist $k \in \mathbb{N}$ and $w_1, w_2, \ldots, w_k \in \{0,1\}^*$ such that $T \subseteq w_1^* w_2^* \cdots w_k^*$. By Lemma 1, if we can establish that there is an infinite T'-code, where $T' = w_1^* \cdots w_k^*$, the result will follow. Thus, without loss of generality, we let $T = w_1^* w_2^* \cdots w_k^*$.

If $w_1 = w_2 = \cdots = w_k = \epsilon$, then $T = \{\epsilon\}$, and thus $\mathcal{P}_T(\Sigma) = 2^{\Sigma^+} - \emptyset$, which clearly contains an infinite language.

Otherwise, there exists i_0 with $1 \leq i_0 \leq k$ such that $w_{i_0} \neq \epsilon$. For all $1 \leq i \leq k$, let $\alpha_i = |w_i|$. Let $a, b \in \Sigma$ be distinct letters, and define $L_T \subseteq \{a,b\}^+$ by

$$L_T = \{(\prod_{i=1}^{k} a^m b^{\alpha_i}) a^m \ : \ m \geq 0\}.$$

We have that $L_T \subseteq \{a,b\}^+$ as $\alpha_{i_0} \neq 0$. We claim $L_T \in \mathcal{P}_T(\Sigma)$. Assume not. Then there exist $m_1, m_2 \in \mathbb{N}$ with $m_1 > m_2$, $t \in T$ and $z \in \Sigma^+$ such that

$$(\prod_{i=1}^{k} a^{m_1} b^{\alpha_i}) a^{m_1} \in (\prod_{i=1}^{k} a^{m_2} b^{\alpha_i}) a^{m_2} \sqcup\!\sqcup_t z.$$

Thus, we have that $z = a^{(k+1)(m_1 - m_2)}$. Further, let $t_i \in \{0,1\}^*$ for $1 \leq i \leq k+1$ be defined so that

$$t = (\prod_{i=1}^{k} t_i 0^{\alpha_i}) t_{k+1},$$

where $|t_i|_0 = m_2$ and $|t_i|_1 = m_1 - m_2$ for all $1 \leq i \leq k+1$. As $t \in T$, there exist $j_i \in \mathbb{N}$ for all $1 \leq i \leq k$ such that $t = \prod_{i=1}^{k} w_i^{j_i}$. Thus, we have that

$$\sum_{i=1}^{k} \alpha_i j_i = (\sum_{i=1}^{k} |t_i| + \alpha_i) + |t_{k+1}|,$$

and so

$$\sum_{i=1}^{k} \alpha_i j_i \geq \sum_{i=1}^{k} |t_i| + \alpha_i.$$

Let ℓ with $1 \leq \ell \leq k$ be the minimal index such that

$$\sum_{i=1}^{\ell} \alpha_i j_i \geq \sum_{i=1}^{\ell} |t_i| + \alpha_i. \tag{8}$$

Note that $j_\ell > 0$, since if $j_\ell = 0$, then $\ell - 1$ satisfies (8) as well, contrary to our choice of ℓ (if $\ell = 1$ and $j_1 = 0$ then $|t_1| = 0$, which is a contradiction to $|t_1| = m_1$).

Let $u_1 = \prod_{i=1}^{\ell-1} t_i 0^{\alpha_i}$, $u_2 = (\prod_{i=\ell+1}^{k} t_i 0^{\alpha_i}) t_{k+1}$, $s_1 = \prod_{i=1}^{\ell-1} w_i^{j_i}$ and $s_2 = \prod_{i=\ell+1}^{k} w_i^{j_i}$. Thus, we have that

$$u_1 t_\ell 0^{\alpha_\ell} u_2 = s_1 w_\ell^{j_\ell} s_2$$

with $|u_1| \geq |s_1|$ and $|u_1| + |t_\ell| + \alpha_\ell \leq |s_1| + \alpha_\ell \cdot j_\ell$. The situation is summarized in Fig. 1. Thus, we have that $w_\ell^{j_\ell}$ contains a block of

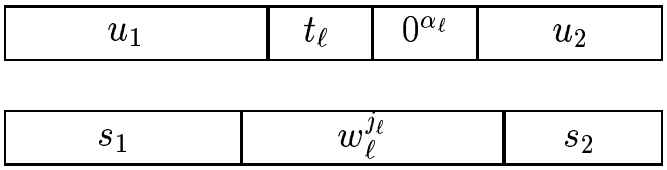

Fig. 1 Two factorizations of t.

zeroes of length α_ℓ. As $j_\ell \neq 0$, this implies that $w_\ell = 0^{\alpha_\ell}$. But then as t_ℓ is a factor of $w_\ell^{j_\ell}$, we also have that $t_\ell \in 0^*$. Thus, $|t_\ell|_1 = 0$, and $m_1 = m_2$, a contradiction. □

Further, there exist uncountably many unbounded trajectories T such that $\mathcal{P}_T$ contains infinite–even infinite *regular*–languages. Infinitely many of these are unbounded regular sets of trajectories.

Theorem 10. *Let $T \subseteq \{0,1\}^*$ be a set of trajectories such that there exists $n \geq 0$ such that $T \subseteq 0^{\leq n}1(0+1)^*$. Then for all Σ with $|\Sigma| > 1$, $\mathcal{P}_T(\Sigma)$ contains an infinite regular language.*

Proof. Let $n \geq 0$ and $T^{(n)} = 0^{\leq n}1(0+1)^*$. By Lemma 1, it suffices to prove that $\mathcal{P}_{T^{(n)}}(\Sigma)$ contains an infinite regular language. Let $a, b \in \Sigma$ be distinct letters. Consider the regular language $R_n = a^{n+1}b^*$.

Assume that $R_n \notin \mathcal{P}_{T^{(n)}}(\Sigma)$. Thus, there exist $i \geq 0$, $t_0 \in T^{(n)}$ and $z \in \{a,b\}^+$ such that $(a^{n+1}b^i \sqcup\!\sqcup_{t_0} z) \cap R_n \neq \emptyset$. Let $t_0 = 0^m 1 t_2$ for some $n \geq m \geq 0$ and $t_2 \in \{0,1\}^*$. Consider that

$$a^{n+1}b^i \sqcup\!\sqcup_{t_0} z = a^m z_1 (a^{n+1-m} b^i \sqcup\!\sqcup_{t_2} z_2)$$

where $z = z_1 z_2$ and $z_1 \in \{a,b\}$.

By assumption, $a^m z_1 (a^{n+1-m} b^i \sqcup\!\sqcup_{t_2} z_2) \cap R_n \neq \emptyset$, so that $z_1 = a$. But now, $(a^{n+1-m} b^i \sqcup\!\sqcup_{t_2} z_2) \cap a^{n-m} b^* \neq \emptyset$, which is clearly impossible, since $|x|_a \geq n+1-m$ for all $x \in a^{n+1-m} b^i \sqcup\!\sqcup_{t_2} z_2$. $\square$

The following corollary holds by Lemma 14.

Corollary 12. *Let $T \subseteq \{0,1\}^*$ be a set of trajectories such that there exists $n \geq 0$ such that $T \subseteq (0+1)^* 10^{\leq n}$. Then for all Σ with $|\Sigma| > 1$, $\mathcal{P}_T(\Sigma)$ contains an infinite regular language.*

We now turn to defining sets T of trajectories such that all T-codes are finite. The following proof is generalized from the case $H = (0+1)^*$ found in, e.g., Lothaire [34] or Conway [4, pp. 63–64].

Lemma 25. *Let $n, m \geq 1$ be such that $m \mid n$. Let $T_{n,m} = (0^n + 1^m)^* 0^{\leq n-1}$. Then $T_{n,m} \in \mathfrak{F}_H$.*

Proof. In what follows, let $\omega = \omega_{T_{n,m}}$. Assume that there exists an infinite $T_{n,m}$-code. Then there exists an infinite sequence $\{x_i\}_{i \geq 1}$ which is ω-free, i.e., $i < j$ implies $x_i \not\omega\, x_j$. As $T_{n,m} \supseteq 0^*$, ω is reflexive and we have that $x_i \neq x_j$ for all $i > j \geq 1$.

We now choose (using the axiom of choice) a minimal infinite ω-free sequence as follows: let y_1 be the shortest word which begins an infinite ω-free sequence. Let y_2 be the shortest word such that y_1, y_2 begins an infinite ω-free sequence. We continue in this way. Let $\{y_i\}_{i \geq 1}$ be the resulting sequence. Clearly, $\{y_i\}_{i \geq 1}$ is an infinite ω-free sequence.

As ω is reflexive, $y_i \neq y_j$ for all $i > j \geq 1$. Therefore, $|y_i| \leq n$ for only finitely many $i \in \mathbb{N}$. Furthermore, since there are only finitely many words of length n, there exist $y \in \Sigma^n$ and $\{i_j\}_{j \geq 1} \subseteq \mathbb{N}$ such that y is a prefix of y_{i_j} for all $j \geq 1$. In particular, for all $j \geq 1$, let $u_j \in \Sigma^*$ be the word such that $y_{i_j} = y u_j$. Consider the sequence

$$Y = \{y_1, y_2, y_3, \cdots, y_{i_1 - 1}, u_1, u_2, \cdots\}.$$

Clearly, as $n \geq 1$, $|u_1| < |y_{i_1}|$. Thus, Y is an infinite sequence which comes before $\{y_i\}_{i \geq 1}$ in our ordering of infinite ω-free sequences, and so two words in Y must be comparable under ω. By assumption, $y_{j_1} \not\omega\, y_{j_2}$ for all $1 \leq j_1 < j_2 \leq i_1 - 1$. Thus, there are two remaining cases:

(i) there exist $1 \leq j \leq i_1 - 1$ and $k \geq 1$ such that $y_j \, \omega \, u_k$. Thus, let $t \in T_{n,m}$ and $\alpha \in \Sigma^*$ be chosen so that $u_k \in y_j \sqcup_t \alpha$. Consider $t' = 1^n t \in T_{n,m}$. Then $y_{i_k} = yu_k \in y(y_j \sqcup_t \alpha) = y_j \sqcup_{t'} y\alpha$. Therefore, $y_j \, \omega \, y_{i_k}$. As $j \leq i_1 - 1 < i_k$, this is a contradiction.

(ii) there exist $k > \ell \geq 1$ such that $u_\ell \, \omega \, u_k$. Let $\alpha \in \Sigma^*$ and $t \in T_{n,m}$ be such that $u_k \in u_\ell \sqcup_t \alpha$. Consider $t' = 0^n t \in T_{n,m}$. Then $y_{i_k} = yu_k \in y(u_\ell \sqcup_t \alpha) = yu_\ell \sqcup_{t'} \alpha = y_{i_\ell} \sqcup_{t'} \alpha$. Thus $y_{i_\ell} \, \omega \, y_{i_k}$. As $\ell < k$, this is a contradiction.

We have arrived at a contradiction. □

As another class of examples, Ehrenfeucht *et al.* [8, p. 317] note that $\{1^n, 0\}^* \in \mathfrak{F}_H$ for all $n \geq 1$ (their other results, though elegant and interesting, do not otherwise seem to be applicable to our situation).

Note that $T_{1,1} = \{0,1\}^*$. Let $T_n = T_{n,n}$. For all $1 \leq i < j$, $\mathcal{P}_{T_i} \neq \mathcal{P}_{T_j}$, as $0^i 1^i \in T_i - T_j$. Thus, by Lemma 2, the classes of T_i- and T_j-codes are distinct.

Corollary 13. *There are infinitely many $T \subseteq \{0,1\}^*$ which define distinct classes $\mathcal{P}_T$ satisfying $\mathcal{P}_T \subseteq$ FIN.*

Further, the following is immediate:

Corollary 14. *Let $T \subseteq \{0,1\}^*$ be such that $T_n \subseteq T$ for some $n \geq 1$. Then $\mathcal{P}_T \subseteq$ FIN.*

Ilie [18, Sect. 7.7] also gives a class of partial orders which we may phrase in terms of sets of trajectories. In particular, define the set of functions

$$\mathcal{G} = \{g : \mathbb{N} \to \mathbb{N} \ : \ g(0) = 0 \text{ and } 1 \leq g(n) \leq n \text{ for all } n \geq 1\}.$$

Then for all $g \in \mathcal{G}$, we define

$$T_g = \{1^* \prod_{k=1}^{m} (0^{i_k} 1^*) \ : \ i_k \geq 0 \ \forall 1 \leq k \leq m; \ m = g(\sum_{k=1}^{m} i_k)\}.$$

We denote the *upper limit* of a sequence $\{s_n\}_{n \geq 1}$ by $\overline{\lim}_{n \to \infty} s_n$. We have the following result [18, Thm. 7.7.8]:

Theorem 11. *Let $g \in \mathcal{G}$. Then $T_g \in \mathfrak{F}_H \iff \overline{lim}_{n \to \infty} \frac{n}{g(n)} < \infty$.*

8.4 Decidability and Finiteness Conditions

We now consider decidability of membership in $\mathcal{P}_T$ if T satisfies the conditions of the previous sections. We have the following positive decidability results:

Theorem 12. *Let T be recursive. If $T \in \mathfrak{F}_R$ (resp., $T \in \mathfrak{F}_C$, $T \in \mathfrak{F}_H$) then given a regular (resp., context-free, context-free) language L, it is decidable whether $L \in \mathcal{P}_T$.*

Proof. We establish the result for $T \in \mathfrak{F}_C$. The case $T \in \mathfrak{F}_H$ is an instance of this case and the case $T \in \mathfrak{F}_R$ is very similar. Let $T \in \text{REC}$ and $T \in \mathfrak{F}_C$. Let $L \in \text{CF}$. We first check if L is infinite. If it is, then certainly $L \notin \mathcal{P}_T$, so we answer no.

If L is finite, then we can effectively find a list of all words in L (consider putting L in CNF). Let $F = L$, where F is some effectively given finite set. Then by Lemma 4, we can decide whether $L = F \in \mathcal{P}_T$. □

One might hope for an undecidability result of the following type, which would complement Theorem 3: given a fixed $T \in \text{REG}$ (perhaps with some reasonable assumption, e.g., completeness), then it is undecidable, given a CFL L, whether $L \in \mathcal{P}_T$. Theorem 12 shows us that we cannot hope for a simple such result, since we need to restrict ourselves to those T which do not lie in $\mathfrak{F}_C$ in this case. It is an open problem to determine suitable conditions on $T \in \text{REG}$ such that the problem of determining membership in $\mathcal{P}_T$ for CFLs is undecidable.

8.5 Up and Down Sets

Let $L \subseteq \Sigma^*$ and $T \subseteq \{0,1\}^*$ Let $\text{DOWN}_T(L), \text{UP}_T(L)$ as

$$\text{DOWN}_T(L) = L \rightsquigarrow_{\tau(T)} \Sigma^*;$$
$$\text{UP}_T(L) = L \sqcup\!\sqcup_T \Sigma^*.$$

Our notation roughly follows Harju and Ilie [13], where $\text{DOWN}_T(L)$ is denoted $\text{DOWN}_{\omega_T}(L)$ and $\text{UP}_T(L)$ is denoted $\text{DOWN}_{\omega_T^{-1}}(L)$.

Our aim in this section is, given T, to characterize the complexity $\text{UP}_T(L)$ and $\text{DOWN}_T(L)$ for arbitrary L. We will have a particular interest in those $T \in \mathfrak{F}_H$ which are partial orders. Let $\mathfrak{F}_H^{(po)}$ denote the class of all trajectories T in $\mathfrak{F}_H$ which are partial orders.

Haines [12] observed that for $T = (0+1)^*$, $\text{UP}_T(L)$ and $\text{DOWN}_T(L)$ are regular languages for all L. There is an elegant generalization of

Haines' result due to Harju and Ilie [13]: If we restrict our attention to those T in $\mathfrak{F}_H$ which are compatible, then $\text{UP}_T(L)$ and $\text{DOWN}_T(L)$ are still regular languages for all languages L. We recall this in the following result, which is a specific case of a result due to Harju and Ilie [13, Thm. 6.3]:[2]

Theorem 13. *Let $T \in \mathfrak{F}_H$ be compatible. Let $L \subseteq \Sigma^*$ be a language. Then $\text{UP}_T(L), \text{DOWN}_T(L)$ are regular languages.*

The following corollary is an interesting consequence:

Corollary 15. *Let $T \in \mathfrak{F}_H$ satisfy $0^* \subseteq T^*$. Let $L \subseteq \Sigma^*$ be a language. Then $\text{UP}_{T^*}(L), \text{DOWN}_{T^*}(L)$ are regular languages.*

Proof. If $0^* \subseteq T^*$ then T^* is clearly compatible by Corollary 6. Further, as $T \subseteq T^*$, we have $T^* \in \mathfrak{F}_H$. The result now follows by Theorem 13. □

We now consider arbitrary $T \in \mathfrak{F}_H^{(po)}$ and seek to characterize the complexity of $\text{UP}_T(L), \text{DOWN}_T(L)$. By the same proofs as given for $H = (0 + 1)^*$ (see, e.g., Harrison [14, Sect. 6.6]), we have the following results:

Lemma 26. *Let $T \subseteq \mathfrak{F}_H^{(po)}$. Let $L \subseteq \Sigma^*$. Then*

(a) there exists a finite language $F \subseteq \Sigma^*$ such that $\text{UP}_T(L) = \text{UP}_T(F)$.
(b) there exists a finite language $G \subseteq \Sigma^*$ such that $\text{DOWN}_T(L) = \overline{\text{UP}_T(G)}$.

Let $\mathcal{C}_1, \mathcal{C}_2$ be classes of languages. Then let $\mathcal{C}_1 \wedge \mathcal{C}_2 = \{L_1 \cap L_2 : L_i \in \mathcal{C}_i, i = 1, 2\}$ and co-$\mathcal{C}_1 = \{\overline{L} : L \in \mathcal{C}\}$.

We now characterize the complexity of $\text{UP}_T(L)$ and $\text{DOWN}_T(L)$ for all L, based on the complexity of T:

Theorem 14. *Let $\mathcal{C}$ be a cone. Let $T \in \mathfrak{F}_H^{(po)}$ be an element of $\mathcal{C}$. Then for all $L \subseteq \Sigma^*$, $\text{UP}_T(L) \in \mathcal{C}$ and $\text{DOWN}_T(L) \in$ co-$\mathcal{C}$.*

Proof. Let $L \subseteq \Sigma^*$. Then there exists $F \subseteq \Sigma^*$ such that $\text{UP}_T(L) = \text{UP}_T(F) = F \sqcup\!\sqcup_T \Sigma^*$. By the closure properties of cones under $\sqcup\!\sqcup_T$, $\text{UP}_T(L) \in \mathcal{C}$. A similar proof shows that $\text{DOWN}_T(L) \in$ co-$\mathcal{C}$. □

[2] Note that what Harju and Ilie call monotone, we call compatible.

8.6 T-Convexity and Maximality Revisited

We now turn to the complexity of T-convex languages:

Theorem 15. *Let $\mathcal{C}$ be a cone. Let $T \in \mathfrak{F}_H^{(po)}$ be an element of $\mathcal{C}$. Then every T-convex language is an element of $\mathcal{C} \wedge$ co-$\mathcal{C}$.*

Proof. Let $T \in \mathfrak{F}_H^{(po)}$. As T is a partial order, it is reflexive. Thus, if L is a T-convex language, we have that $L = \text{UP}_T(L) \cap \text{DOWN}_T(L)$ [7, Cor. 4.1]. Thus, by Theorem 14, the result follows. □

The following corollary is immediate, based on the closure properties of the recursive and regular languages:

Corollary 16. *Let $T \in$ REG (resp., REC) be such that $T \in \mathfrak{F}_H^{(po)}$. If L is a T-convex language, then $L \in$ REG (resp., REC).*

Corollary 16 was known for the case of $H = (0+1)^*$ and $L \in$ REG, see Thierrin [41, Cor. to Prop. 3]. Further, we can also establish the following result:

Theorem 16. *Let $T \in \mathfrak{F}_H$ be compatible. Then every T-convex language is regular.*

Consider the sets $E_n = \{0, 1^n\}^*$. As noted by Ehrenfeucht *et al.* [8], $E_n \in \mathfrak{F}_H$. As $E_n = E_n^*$ and $0^* \subseteq E_n$, E_n is compatible. Thus, we have that every E_n-convex language is regular.

9 Conclusions

We have introduced the notion of a T-code, and examined its properties. Many results which are known in the literature are specific instances of general results on T-codes. However, the notion of a T-code is not so general as to prevent interesting results from being obtained. We feel that the framework of T-codes is very suitable for further analysis of the general structure of the many classes of codes which it generalizes. Further research into this area should prove very useful.

Acknowledgements. Many thanks for Kai Salomaa for his careful reading of this paper. The help of the anonymous referee, who made several helpful suggestions which improved the presentation of this paper, is also gratefully acknowledged.

References

1. J. Berstel and D. Perrin. *Theory of Codes.* Available at http://www-igm.univ-mlv.fr/%7Eberstel/LivreCodes/Codes.html, 1996.
2. V. Bruyère and D. Perrin. Maximal bifix codes. *Theor. Comput. Sci.*, 218:107–121, 1999.
3. C. Choffrut and J. Karhumäki. Combinatorics on words. pages 329–438. In [36].
4. J. Conway. *Regular Algebra and Finite Machines.* Chapman and Hall, 1971.
5. A. de Luca and S. Varricchio. Regularity and finiteness conditions. pages 747–810. In [36].
6. M. Domaratzki. Deletion along trajectories. *Accepted, Theor. Comp. Sci.*, 2004.
7. M. Domaratzki. Trajectory-based embedding relations. *Accepted, Fund. Inf.*, 2004.
8. A. Ehrenfeucht, D. Haussler, and G. Rozenberg. On regularity of context-free languages. *Theor. Comput. Sci.*, 23:311–332, 1983.
9. P. Flajolet and J.-M. Steyaert. On sets having only hard subsets. In J. Loeckx, editor, *Automata Languages and Programming*, volume 14 of *Lecture Notes in Computer Science*, pages 446–457, 1974.
10. S. Ginsburg. *The Mathematical Theory of Context-Free Languages.* McGraw-Hill, 1966.
11. Y. Guo, H. Shyr, and G. Thierrin. E-Convex infix codes. *Order*, 3:55–59, 1986.
12. L. Haines. On free monoids partially ordered by embedding. *J. Comb. Theory*, 6:94–98, 1969.
13. T. Harju and L. Ilie. On quasi orders of words and the confluence property. *Theor. Comput. Sci.*, 200:205–224, 1998.
14. M. Harrison. *Introduction to Formal Language Theory.* Addison-Wesley, 1978.
15. G. Higman. Ordering by divisibility in abstract algebras. *Proc. Lond. Math. Soc.*, 2(3):326–336, 1952.
16. J. E. Hopcroft and J. D. Ullman. *Introduction to Automata Theory, Languages, and Computation.* Addison-Wesley, 1979.
17. L. Ilie. Remarks on well quasi orders of words. In S. Bozapalidis, editor, *Proceedings of the 3rd DLT*, pages 399–411, 1997.
18. L. Ilie. *Decision Problems on Orders of Words.* PhD thesis, University of Turku, 1998.
19. L. Ilie and A. Salomaa. On well quasi orders of free monoids. *Theor. Comput. Sci.*, 204:131–152, 1998.
20. M. Ito, H. Jürgensen, H. Shyr, and G. Thierrin. Outfix and infix codes and related classes of languages. *J. Comput. Syst. Sci.*, 43:484–508, 1991.
21. P. Jullien. Sur un théorème d'extension dans la théorie des mots. *C.R. Acad. Sci., Paris, Sér. A*, 266:851–854, 1968.
22. H. Jürgensen and S. Konstantinidis. Codes. pages 511–600. In [36].
23. H. Jürgensen, H. Shyr, and G. Thierrin. Codes and compatible partial orders on free monoids. In S. Wolfenstein, editor, *Algebra and Order: Proceedings of the First International Symposium on Ordered Algebraic Structures, Luminy–Marseilles 1984*, pages 323–334. Heldermann Verlag, 1986.
24. H. Jürgensen and S. S. Yu. Relations on free monoids, their independent sets, and codes. *Int. J. Comput. Math.*, 40:17–46, 1991.

25. A. Kadrie, V. Dare, D. Thomas, and K. Subramanian. Algebraic properties of the shuffle over ω-trajectories. *Inf. Process. Letters*, 80(3):139–144, 2001.
26. L. Kari. On language equations with invertible operations. *Theor. Comput. Sci.*, 132:129–150, 1994.
27. L. Kari, S. Konstantinidis, and P. Sosík. On properties of bond-free DNA languages. Technical Report 609, Computer Science Department, University of Western Ontario, 2003. Submitted for publication.
28. L. Kari and P. Sosík. Language deletions on trajectories. Technical Report 606, Computer Science Department, University of Western Ontario, 2003. Submitted for publication.
29. L. Kari and G. Thierrin. k-catenation and applications: k-prefix codes. *J. Inf. Optimization Sci.*, 16(2):263–276, 1995.
30. J. Kruskal. The theory of well-quasi-ordering: A frequently discovered concept. *J. Comb. Theory, Ser. A*, 13:297–305, 1972.
31. N. Lam. Finite maximal infix codes. *Semigroup Forum*, 61:346–356, 2000.
32. D. Long. On two infinite hierarchies of prefix codes. In K. Shum and P. Yuen, editors, *Proceedings of the Conference on Ordered Structures and Algebra of Computer Languages*, pages 81–90. World Scientific, 1993.
33. D. Long. k-bifix codes. *Rivista di Matematica Pura ed Applicata*, 15:33–55, 1994.
34. M. Lothaire. *Combinatorics on Words.* Addison-Wesley, 1983.
35. A. Mateescu, G. Rozenberg, and A. Salomaa. Shuffle on trajectories: Syntactic constraints. *Theor. Comput. Sci.*, 197:1–56, 1998.
36. G. Rozenberg and A. Salomaa, editors. *Handbook of Formal Languages, Vol. I.* Springer-Verlag, 1997.
37. H. Shyr. *Free Monoids and Languages.* Hon Min Book Company, Taichung, Taiwan, 2001.
38. H. Shyr and G. Thierrin. Hypercodes. *Inf. Control*, 24(1):45–54, 1974.
39. H. Shyr and G. Thierrin. Codes and binary relations. In A. Dold and B. Eckmann, editors, *Séminaire d'Algébre Paul Dubreil, Paris 1975–1976*, volume 586 of *Lecture Notes in Mathematics*, pages 180–188. Springer-Verlag, 1977.
40. N. Sloane. *The On-Line Encyclopedia of Integer Sequences.* Published electronically at http://www.research.att.com/~njas/sequences, 2004.
41. G. Thierrin. Convex languages. In M. Nivat, editor, *Automata, Languages and Programming*, pages 481–492. North-Holland, Amsterdam, 1972.
42. G. Thierrin and S.S. Yu. Shuffle relations and codes. *J. Inf. Optimization Sci.*, 12(3):441–449, 1991.
43. J. van Leeuwen. Effective constructions in well-partially ordered free monoids. *Discrete Math.*, 21:237–252, 1978.
44. S. Yu. Regular languages. pages 41–110. In [36].
45. L. Zhang and Z. Shen. Completion of recognizable bifix codes. *Theor. Comput. Sci.*, 145:345–355, 1995.