An algorithmic approach to the problem of a semiretract base

Wit Foryś*, Tomasz Krawczyk

Institute of Computer Science, Jagiellonian University, Nawojki 11, 30-072 Krakow, Poland

Received 22 February 2005; received in revised form 20 May 2006; accepted 14 September 2006

Communicated by M. Ito

Abstract

A semiretract of a free monoid $A^*$ is an intersection of a family of retracts of $A^*$ and it is a free submonoid. In the paper we propose an algorithmic approach to the problem of finding the base (code) of a semiretract.

© 2006 Elsevier B.V. All rights reserved.

Keywords: Free monoid; Retract; Semiretract; Key-code; Algorithm

1. Introduction

The notion of a semiretract was introduced by Anderson [1] and was the subject of a research in [2–11]. In [3] Anderson characterized semiretracts in terms of codes—generator sets. Unfortunately, the main theorem of the paper appeared to be false and the problem was undertaken and revised in our paper [10]. Besides this error the paper [3] contains the basic facts in our consideration, namely that any semiretract is an intersection of a family of retracts generated by codes having the common set of keys. In this paper we propose an algorithmic approach to the problem of finding the base (code) of a semiretract. We believe that our investigations also cast a new light on the inner structure of semiretracts.

2. Basic notions and definitions

We assume the reader is familiar with the basic notions and concepts from the theories of semigroups, automata and formal languages [13].

Let $A$ be any finite set and let $A^*$ denote a free monoid generated by $A$. The length of a word $w \in A^*$, in symbols $|w|$, is defined to be the number of letters occurring in $w$ (the length of the empty word $1$ equals $0$).

A retraction $r : A^* \to A^*$ is a morphism for which $r \circ r = r$. A retract of $A^*$ is the image of $A^*$ by a retraction. A semiretract of $A^*$ is the intersection of a family of retracts of $A^*$. A word $w \in A^*$ is called a key-word if there is at least one letter in $A$ that occurs exactly once in $w$. This letter is called a key of $w$. A set $C \subseteq A^*$ of key-words is
Theorem 2.1. \( R \subset A^* \) is a retract of \( A^* \) if and only if \( R = C^* \), where \( C \) is a key-code.

In [3] Anderson proved the following theorem, basic for our considerations.

Theorem 2.2. Let \( S = \bigcap_{i=1}^{n} C_i^* \) be a semiretract given by key-codes \( C_i \subset A^* \) for \( i = 1, \ldots, n \). There exist key-codes \( D_i \subset A^* \) such that
1. \( S = \bigcap_{i=1}^{n} D_i^* \) and \( D_i \subset C_i^* \) for \( i = 1, \ldots, n \),
2. \( \text{key}(D_1) = \text{key}(D_2) = \cdots = \text{key}(D_n) \).

Hence, any semiretract \( S \) is an intersection of a family of retracts generated by key-codes having the common set of keys.

In the paper, we present an algorithmic approach to the assertion of the theorem which leads to the construction of a finite automaton that recognizes the base of a semiretract. We assume, throughout the paper that any semiretract of keys \( w_i \) where \( i \in \mathbb{N} \) to the possibility of tiling the table \( V \) by \( C_i \) for \( i \in \mathbb{N} \). In the sequel any injection \( \text{key} : C \to A \) is called key-injection. Given a key-code \( C \) and a fixed key-injection \( \text{key} \) the set of all keys of words in \( C \) is denoted by \( \text{key}(C) \).

The following characterization of retracts is due to Head [12].

Any domino in the table \( T(w) \) is identified by the triple \( (i, u, x) \) where \( i \in \{1, \ldots, n\} \), \( u \in C_i \) and \( x \in \{1, \ldots, |w|\} \) points out a position of \( \text{key}(u) \) in \( T(w) \) counted from the beginning of the row. Notice that, if \( u = a_{-p} \ldots a_0 \ldots a_r \) then the domino \((i, u, x)\) covers the entries (squares) from the set \( \{(i, j) \mid x - p \leq j \leq x + r\} \). The set of squares covered by a domino \((i, u, x)\) is denoted by \( \{(i, u, x)\} \).
3. Results

Definition 2.3. Let \( A \subset V \times \mathbb{Z} \). The \( A \) is a configuration of dominoes if \( A \) fulfills the following conditions:

1. the set of squares \( \{(i, u, x) \mid (i, u, x) \in A\} \) consists of pairwise disjoint elements for any \( i \in \{1, \ldots, n\} \) (the dominoes do not overlap in the table \( A \)),
2. for any fixed \( j \in \mathbb{Z} \) and \( i_1, i_2 \in \{1, \ldots, n\} \) there are no two squares \((i_1, j)\) and \((i_2, j)\) in \( A \) filled up with two different letters.

Of course if \( w \) is in \( S \) then a table \( T(w) \) is a configuration. For a configuration of dominoes \( A \) we denote by

- \( A^{\rightarrow z} \) the shift of \( A \) by \( z \in \mathbb{Z} \), that is,
  \[
  A^{\rightarrow z} = \{(i, u, x + z) \mid (i, u, x) \in A\}.
  \]
- \([A]\) the set of squares covered by dominoes from \( A \),
  \[
  [A] = \bigcup \{(i, u, x) \mid (i, u, x) \in A\}.
  \]

Note that in a configuration \( A \) dominoes do not overlap one another but there are possible gaps between them. We say that a configuration \( A \) is connected if for any \( i \in \{1, \ldots, n\} \) the set \( \{x \in \mathbb{Z} \mid (i, x) \in [A]\} \) is equal to \([p, q]\) for some \( p, q \in \mathbb{Z} \).

3. Results

Let us consider a domino \((i, u, x) \in T(w), w \in S\). Assume that for some \( j \neq i \) there exists \( v \in C_j \) such that the key of \( v \) occurs in the word \( u \). Hence, there exists a square \((i, z) \in \{(i, u, x)\}\) which is filled up with the letter key\((v)\). Since key\((v)\) occurs only once in the word \( v \in C_j \) and in no other word from \( C_j \), then the only domino in \( j \)-th row which can cover the square \((j, z)\) with a letter key\((v)\) is \((j, v, z)\). Hence, the element \((j, v, z)\) has to occur in \( T(w) \). In general, a domino \((i, u, x) \in T(w)\) enforces in all other rows an occurrence of these dominoes that have as key-letters the letters occurring in \( u \). To obtain a clear cut picture of those dependencies we introduce a relation \( E \) on dominoes and a labeled multidigraph associated with \( S \).

Definition 3.1. Let \( S = \bigcap_{i=1}^{n} C_i^{*} \) be a semiretract, \( C_1, \ldots, C_n \subset A^{*} \) key-codes. Let \( V \) be the set of all dominoes of \( S \) and \((i, u), (j, v) \in V \) two dominoes such that \( u = a_{-p} \ldots a_0 \ldots a_r \) and \( v = b_{-s} \ldots b_0 \ldots b_l \). A triple \(((i, u), z, (j, v))\) is in the relation \( E \subset V \times \mathbb{Z} \times V \) if and only if \( a_z = b_0 \) for some \( z \in \{-p, \ldots, 0, \ldots, r\} \).

We consider relation \( E \) as the set of arrows between nodes and dominoes in \( V \) labeled by integers. We use in the sequel the notation \((i, u) \rightarrow_z (j, v)\) for a triple \(((i, u), z, (j, v))\) in \( E \) and say that \((i, u)\) enforces \((j, v)\).

Definition 3.2. Let \( S = \bigcap_{i=1}^{n} C_i^{*} \) be a semiretract, \( C_1, \ldots, C_n \subset A^{*} \) key-codes. A directed multigraph \( G = (V, E) \) where \( V = \bigcup_{i=1}^{n} (i) \times C_i \) is the set of all dominoes and \( E \) considered as a relation on \( V \) with integer labels is called a labeled multidigraph associated with \( S \). For \((i_0, v_0), (i_m, v_m) \in V \) we say that there is a path \((i_0, v_0) \rightarrow_x^{*} (i_m, v_m)\)
in $G = (V, E)$ if there exist nodes $(i_1, v_1), \ldots, (i_m, v_m) \in V$ and integers $x_1, \ldots, x_m \in \mathbb{Z}$ such that

1. $(i_0, v_0) \to x_1 (i_1, v_1) \to x_2 \cdots \to x_m (i_m, v_m)$,
2. $\sum_{i=1}^{m} x_i = x$.

Using the above introduced notions we can reword the observation done at the beginning of this section as follows.

**Fact 3.3.** If $(i, u, x)$ is in $T(w)$ for some $w \in S$, $x \in \{1, \ldots, |w|\}$ and $(i, u) \to_{z}^{*} (j, v)$ for some $z \in \mathbb{Z}$, then $(j, v, x + z)$ is in $T(w)$.

This fact motivates the following definition.

**Definition 3.4.** Let $S = \bigcap_{i=1}^{n} C_{i}^{x}$ be a semiretract, $C_1, \ldots, C_n \subset A^{x}$ key-codes, $w \in S$ and $(i, u, x) \in T(w)$. The set

$$B(i, u, x) = \{(j, v, x + z) \in T(w) | (i, u) \to_{z}^{*} (j, v), (j, v) \in V, z \in \mathbb{Z}\}$$

is called a neighborhood of $(i, u, x)$ in $T(w)$.

Hence $B(i, u, x)$ is a part of the table $T(w)$ that contains the domino $(i, u, x)$ itself and all dominoes in $T(w)$ that are enforced by it.

Let us denote by $CC(G)$ the set of all strongly connected components $W \subset V$ in a multidigraph $G = (V, E)$. For any $W \in CC(G)$ we fix a node $(i, u) \in W$ and we call this node a representant of $W$. To express the fact that $(i, u)$ represents $W$ we write $W_{(i, u)}$. We also denote by $CC_{S}(G) \subset CC(G)$ the set

$$CC_{S}(G) = \{W_{(i, u)} \in CC(G) | \exists w \in S \exists x \in \{1, \ldots, |w|\}, (i, u, x) \in T(w)\}.$$

For a connected component $W_{(i, u)} \in CC(G)$ we define two sets

$$B(W_{(i, u)}) = \{(j, v, z) | (i, u) \to_{z}^{*} (j, v), (j, v) \in V, z \in \mathbb{Z}\}$$

and

$$Bs(W_{(i, u)}) = \{(j, v, z)(i, u) \to_{z}^{*} (j, v), (j, v) \in W_{(i, u)}, z \in \mathbb{Z}\}$$

that we call the neighborhood of $W_{(i, u)}$ and the base of $W_{(i, u)}$, respectively.

The following lemma contains a description of the base and the neighborhood of a strongly connected component belonging to $CC_{S}(G)$.

**Lemma 3.5.** Let $W_{(i, u)} \in CC_{S}(G)$ be a component in $G = (V, E)$. Then

1. For any dominoes $(j_1, v_1), (j_2, v_2) \in W_{(i, u)}$ there exists exactly one $z \in \mathbb{Z}$ such that $(j_1, v_1) \to_{z}^{*} (j_2, v_2)$.
2. If $(j_1, v_1) \to_{z}^{*} (j_2, v_2)$, then $(j_2, v_2) \to_{z}^{*} (j_1, v_1)$.
3. If $(i, u, x) \in T(w)$ and $x \in \{1, \ldots, |w|\}$, then all neighborhoods of dominoes from $Bs(W_{(i, u)})^{\to_{x}}$ coincide and are equal to $B(W_{(i, u)})^{\to_{x}}$.
4. The base $Bs(W_{(i, u)})$ is a configuration of dominoes, the neighborhood $B(W_{(i, u)})$ is a connected configuration of dominoes.

**Proof.** Since $W_{(i, u)} \in CC_{S}(G)$, then there exists $w \in S$ and $x \in \{1, \ldots, |w|\}$ such that $(i, u, x) \in T(w)$. To prove (1) assume, for the indirect proof that there exist $z_1, z_2 \in \mathbb{Z}$, $z_1 \neq z_2$ such that $(j_1, v_1) \to_{z_1}^{*} (j_2, v_2)$ and $(j_1, v_1) \to_{z_2}^{*} (j_2, v_2)$. Since $(j_1, u_1)$ and $(j_2, v_2)$ belong to the strongly connected component $W_{(i, u)}$, there exists $y \in \mathbb{Z}$ such that $(j_2, v_2) \to_{y}^{*} (j_1, v_1)$. Hence, $(j_1, v_1) \to_{y + z_1}^{*} (j_1, v_1)$ and $(j_1, v_1) \to_{y + z_2}^{*} (j_1, v_1)$. Therefore, there exists $t \neq 0$ such that $(j_1, v_1) \to_{t}^{*} (j_1, v_1)$. Since $(i, u) \to_{x_1}^{*} (j_1, v_1)$ for some $x_1 \in \mathbb{Z}$, then $(j_1, v_1, x + x_1) \in T(w)$ by Fact 3.3. For the same reason the dominoes $(j_1, v_1, x + x_1 + j \cdot t)$ for $j \in \mathbb{Z}$ are pairwise disjoint and belong to $T(w)$. This contradicts the fact that the table $T(w)$ is finite.

Analogically we can prove the second part of (1).

Since $(i, u, x) \in T(w)$, it follows by Fact 3.3 that $Bs(W_{(i, u)})^{\to_{x}} \subset T(w)$ and $B(W_{(i, u)})^{\to_{x}} \subset T(w)$. Then, statement (2) follows easily by definition of the base and the neighborhood of $W_{(i, u)}$. 
As $Bs(W_{(i,u)})^{\rightarrow x}$, $B(W_{(i,u)})^{\rightarrow x} \subset T(w)$ it follows that $Bs(W_{(i,u)})$ and $B(W_{(i,u)})$ as well are configuration of dominoes. By definition, the configuration $B(W_{(i,u)})$ is connected. □

From the above lemma one can derive that for any $W_{(i,u)} \in CC_S(G)$ the position $x \in \{1, \ldots, |w|\}$ of the representant $(i, u)$ in the table $T(w)$, $w \in S$, fixes also the position of the neighborhood $B(W_{(i,u)})^{\rightarrow x}$.

**Definition 3.6.** Let $w \in S$. A strongly connected component $W_{(i,u)}$ occurs in the table $T(w)$ at the position $x$ if and only if $B(W_{(i,u)})^{\rightarrow x} \subset T(w)$.

Let us partially order the sets $CC(G)$ and $CC_S(G)$ putting

$$W_{(i,u)} \sqsubseteq W_{(j,v)} \iff \exists y \in \mathbb{Z}: B(W_{(i,u)})^{\rightarrow y} \subset B(W_{(j,v)})$$

By $maxCC(G)$ and $maxCC_S(G)$ we denote the sets of all maximal elements in $(CC(G), \sqsubseteq)$ and $(CC_S(G), \sqsubseteq)$, respectively.

**Fact 3.7.** Let $w \in S$ and $B(W_{(i,u)})^{\rightarrow x} \subset T(w)$ for some $x \in \{1, \ldots, |w|\}$. If there exists $(i_1, u_1, x_1) \in Bs(W_{(i,u)})$ and $k \in \{1, \ldots, n\}$ such that $(k, x_1) \notin [Bs(W_{(i,u)})]$, then there exists $W_{(i,v)} \notin W_{(i,u)}$ and $y \in \{1, \ldots, |w|\}$ such that $B(W_{(i,u)})^{\rightarrow x} \subset B(W_{(j,v)})^{\rightarrow y} \subset T(w)$. It follows $W_{(i,v)} \in CC_S(G)$ and $W_{(i,u)} \sqsubseteq W_{(j,v)}$.

**Proof.** Since $(k, x_1) \notin Bs(W_{(i,u)})$, then $(k, x + x_1) \notin [Bs(W_{(i,u)})]^{\rightarrow x}$. But $(k, x + x_1) \in [(j_1, v_1, y_1)]$ for some domino $(j_1, v_1, y_1)$ in $T(w)$). Assume that $(j_1, v_1) \in W_{(i,v)}$ for some $W_{(j,v)} \in CC_S(G)$. Then $(j_1, v_1, y_1) \in Bs(W_{(j,v)})^{\rightarrow y}$ for some $y \in \{1, \ldots, |w|\}$ and consequently $B(W_{(i,u)})^{\rightarrow x} \subset B(W_{(j,v)})^{\rightarrow y} \subset T(w)$. If $W_{(i,u)} = W_{(j,v)}$ (that means $(i, u) = (j, v)$, then $(j, v) \mapsto y - x (j, v)$. As $(j, v) \mapsto 0 (j, v)$, we get $x = y$ by Lemma 3.5(1). Hence $(k, x + x_1) \in [Bs(W_{(i,u)})]^{\rightarrow x}$ and $(k, x + x_1) \notin [Bs(W_{(i,u)})]^{\rightarrow x}$. It implies $W_{(i,u)} \neq W_{(i,v)}$. □

From the above fact we get immediately:

**Lemma 3.8.** Let $W_{(i,u)} \in CC_S(G)$, where $G = (V, E)$ is a labeled multigraph associated with a semiretract $S = \bigcap_{i=1}^{n} C_i^*$. Then,

1. $W_{(i,u)} \in maxCC_S(G)$ if and only if for every domino $(i_1, u_1, x_1) \in Bs(W_{(i,u)})$ we have $(k, x_1) \in [Bs(W_{(i,u)})]$ for $k = 1, \ldots, n$.
2. If $W_{(i,u)}$ occurs in $T(w)$ at a position $x$ for some $w \in S$, $x \in \{1, \ldots, |w|\}$, then there exists $W_{(i,v)} \in maxCC_S(G)$ occurring in $T(w)$ at some position $y \in \{1, \ldots, |w|\}$ such that $B(W_{(i,u)})^{\rightarrow x} \subset B(W_{(j,v)})^{\rightarrow y}$.

For any $W_{(i,u)} \in CC_S(G)$ we put

$$r(W_{(i,u)}) = \min_{j \in \{1, \ldots, n\}} \{r_j\}$$

and

$$R(W_{(i,u)}) = \max_{j \in \{1, \ldots, n\}} \{r_j\},$$

where $r_j = \max\{y \in \mathbb{N} : (j, y) \in [B(W_{(i,u)})]\}$ for any $j \in \{1, \ldots, n\}$.

We define $l(W_{(i,u)})$ and $L(B(W_{(i,u)}))$ similarly (see Fig. 2).

**Theorem 3.9.** Let $S = \cap_{i=1}^{n} C_i^*$ be a semiretract given by key-codes $C_i \subset A^*$ for $i = 1, \ldots, n$. There exist key-codes $D_i \subset A^*$ such that

1. $S = \cap_{i=1}^{n} D_i^*$ and $D_i \subset C_i^*$ for $i = 1, \ldots, n$,
2. $key(D_1) = key(D_2) = \cdots = key(D_n)$.

The key-codes $D_1, \ldots, D_n$ are effectively computable.
Fig. 2. The configuration $B(W(i,u))$. The set $B'(W(i,u))$ is included in the bordered area. The squares filled up with a key are distinguished with a black rectangle.

Fig. 3. The table $T(w)$ and the sequence $W(i_1,u_1), W(i_2,u_2), \ldots, W(i_m,u_m)$ of all maximal components occurring in $T(w)$. The sets $B'(W(i_j,u_j)) \rightarrow_{x_j}$ for $j = 1, \ldots, m$ are included in the bordered area.

Proof. Since the assertion is obvious for $S = \{1\}$ let us consider a nontrivial semirectract $S$ and a component $W(i,u) \in \text{maxCC}_S(G)$ represented by $(i, u) \in V$ (Fig. 2). By Lemma 3.5(3), the neighborhood $B(W(i,u))$ is a connected configuration. Denote by $B'(W(i,u))$ the set

$$B'(W(i,u)) = B(W(i,u)) \cup \{B(j, v, z) | (j, v, z) \in B(W(i,u)), r(W(i,u)) < z \leq R(W(i,u))\}.$$ 

By the description of maximal elements in $CC_S(G)$ given in Lemma 3.8(1) we have $B'(W(i,u)) \subset B'(W(i,u))$. Since $B'(W(i,u)) \subset B'(W(i,u))$ and $B'(W(i,u))$ is a connected configuration of dominoes (Lemma 3.5(3)), then $B'(W(i,u))$ is also a connected configuration of dominoes. Let $v_k(W(i,u))$ for $k = 1, \ldots, n$ denote the word contained in the $k$th row of $B'(W(i,u))$. Since $B'(W(i,u))$ is connected, the word $v_k(W(i,u))$ is properly defined and $v_k(W(i,u)) \in C_k^*$ for $k = 1, \ldots, n$. Moreover, the words $v_1(W(i,u)), \ldots, v_n(W(i,u))$ are key-words with a common key-letter—we can choose key of $u$ as a common key.

Now we can easily check for $k = 1, \ldots, n$ that the sets

$$D_k = \{v_k(W(i,u)) | W(i,u) \in \text{maxCC}_S(G)\}$$

are key-codes with common key-set $\text{key}(D_1) = \cdots = \text{key}(D_n)$.

The inclusion $D_k^* \subset C_k^*$ for $k = 1, \ldots, n$ follows by the definition of $D_k$. To complete the proof let us consider a word $w$ in $S$. Let $W(i_1,u_1), \ldots, W(i_m,u_m)$ (Fig. 3) be a sequence of all maximal connected components that occur in $T(w)$ at positions $x_1 < \cdots < x_m$, respectively. By Lemma 3.8(2) we get $T(w) = \bigcup_{j=1}^{m} B(W(i_j,u_j)) \rightarrow_{x_j}$. Since for $j = 1, \ldots, m - 1$ we have

$$B'(W(i_j,u_j)) \rightarrow_{x_j} = B(W(i_j,u_j)) \rightarrow_{x_j} \setminus B(W(i_{j+1},u_{j+1})) \rightarrow_{x_{j+1}},$$

then

$$w = v_1(W(i_1,u_1)) \ldots v_1(W(i_m,u_m)),$$

$$w = v_2(W(i_1,u_1)) \ldots v_2(W(i_m,u_m)),$$

$$\vdots$$

$$w = v_n(W(i_1,u_1)) \ldots v_n(W(i_m,u_m)).$$
It means that \( w \in D_k^* \) for \( k = 1, \ldots, m \). The observation that all operations which are described above and which lead to key-codes \( D_i \) for \( i = 1, \ldots, n \) are executed on a finite labeled directed ultigraph \( G = (V, E) \) finishes the proof of the theorem. \( \square \)

4. An algorithm

The main theorem proved in the previous paragraph points out the importance of strongly connected components in \( CCS(G) \), especially those in \( maxCCS(G) \). On the basis of the results of the previous section \( W_{(i,u)} \in CC(G) \) has to fulfill the following conditions in order to be in \( maxCCS(G) \):

(i) \( Bs(W_{(i,u)}) \) is a configuration of dominoes—Lemma 3.5(3).
(ii) \( B(W_{(i,u)}) \) is a connected configuration of dominoes—Lemma 3.5(3).
(iii) For every domino \( (i_1, u_1, x_1) \in Bs(W_{(i,u)}) \) the column \( x_1 \) is covered by dominoes from \( Bs(W_{(i,u)}) \)—Lemma 3.8(1).
(iv) A component \( W_{(i,u)} \) occurs in a sequence \( W_{(i_1,u_1)}, \ldots, W_{(i_m,u_m)} \) such that

- \( W_{(i_1,u_1)} \) is initial (Lemma 3.1).
- For \( j = 1, \ldots, m-1 \) the set \( B(W_{(i_j,u_j)}) \) is connected configuration of dominoes for some \( x_{j+1} > 0 \).
- \( W_{(i_m,u_m)} \) is final (Lemma 3.9).

Based on the above we can construct a data structure that stores words of newly constructed key-codes \( D_1, \ldots, D_n \) with a common key-set in time

\[
O(\max(n, \log |A|) \cdot (|C_1| + \cdots + |C_n|)),
\]

where \( |C_i| = \sum_{w \in C_i} |w| \) and \( |A| \) is the number of elements in the alphabet over which the codes \( C_1, \ldots, C_n \) are defined. Note that the length of the input is equal to \( |C_1| + \cdots + |C_n| \).

5. A minimal deterministic automaton that recognizes the base of a semiretract \( S \)

The last problem we want to deal with in this paper is a construction of the minimal, deterministic automaton \( A_5 \) that recognizes the base of a semiretract \( S \). Theorem 3.9 establishes also a bijection between the set \( maxCCS(G) \) and a common key-set \( K \) of newly constructed key-codes \( D_1, \ldots, D_n \subset A^* \).

Assume that \( k \in K \). For \( i = 1, \ldots, n \) let us denote by \( v_i(k) \in D_i \) a word with the key \( k \)—it means that \( v_i(k) = l_i(k) r_i(k) \) for some \( l_i(k), r_i(k) \in A^* \). We say that \( k \in K \) is initial (final) if and only if \( l_1(k) = \cdots = l_n(k) \) (\( r_1(k) = \cdots = r_n(k) \)). We say that \( k_2 \in K \) follows \( k_1 \in K \) if and only if \( r_1(k_1) l_1(k_2) = \cdots = r_n(k_1) l_n(k_2) \).

In order to get a characterization of words in the base of a semiretract, let us introduce two equivalence relations \( \lambda \) and \( \rho \) on the set \( K \).

**Definition 5.1.** We say that \( k_1, k_2 \in K \) are in a relation \( \lambda (\rho) \) if and only if there exists \( k \in K \) such that \( k \) follows \( k_1 \) and \( k_2 \) \((k_1 \text{ and } k_2 \text{ follow } k)\). Note that there exists in \( K_{/\rho} \) an equivalence class (block) that consists of all initial keys. This equivalence class is denoted by \( L_{init} \). Dually in \( K_{/\lambda} \) there exists a block that consists of all final keys. This block is denoted by \( R_{final} \).

Suppose that \( K_{/\lambda} = \{ L_{init}, L_1, \ldots, L_m \} \). Hence \( K_{/\rho} = \{ R_{final}, R_1, \ldots, R_m \} \) where \( R_1, \ldots, R_m \subset K \) are such that \( k_2 \) follows \( k_1 \) if and only if \( k_1 \in R_x \) and \( k_2 \in L_x \) for some \( x \in \{1, \ldots, m\} \).

Now we can formulate the following

**Fact 5.2.** If a sequence \( k_1, \ldots, k_m \in K \) is such that

(i) \( k_1 \) is initial,
(ii) \( k_{j+1} \) follows \( k_j \) (it means \( (k_j, k_{j+1}) \subset R_x \times L_x \) for some \( x \in \{1, \ldots, m\} \)) for \( j = 1, \ldots, m-1 \),
(iii) \( k_m \) is final,
then the word
\[ w = l_1(k_1)k_1r_1(k_1) \ldots l_1(k_m)k_mr_1(k_m) = \cdots = l_n(k_1)k_1r_n(k_1) \ldots l_n(k_m)k_mr_n(k_m) \]
is in the base of a semiretract \( S \). Moreover, if \( w \in S \), then there exists a sequence \( k_1, \ldots, k_m \in K \) satisfying (i)–(iii) such that the above equality is true.

Let us choose a pair \( (f_1, f_2) \in R_x \times L_x \), where \( x \in \{1, \ldots, m\} \). Assume that \( r_i(f_1) \) and \( r_j(f_1) \) are, respectively, the longest and the shortest words in \( \{r_1(f_1), \ldots, r_m(f_1)\} \). Then \( r_j(f_1)sepx = r_i(f_1) \) for some word \( sepx \in A^+ \). It is easy to verify that for any pair \( (k_1, k_2) \in R_x \times L_x \) there exist \( right(k_1) \), \( left(k_2) \in A^+ \) such that
\[ r_1(k_1)l_1(k_2) = \cdots = r_n(k_1)l_n(k_2) = right(k_1)sepx, left(k_2). \]

If we put \( right(k) = r_1(k) \) for any \( k \in R_{\text{final}} \) and \( left(k) = l_1(k) \) for any \( k \in L_{\text{init}} \), then the words \( left(k), right(k) \) are defined for any \( k \in K \). Having that we can reformulate Fact 5.2.

**Fact 5.3.** Let a sequence \( k_1, \ldots, k_m \in K \) satisfy (i)–(iii). Then the word
\[ w = left(k_1)k_1right(k_1)sepx_1 \ldots sepx_{p-1}left(k_p)k_pright(k_p), \]
where for \( j = 1, \ldots, p-1 \), \( (k_j, k_{j+1}) \in R_{x_j} \times L_{x_j} \) for some \( x_j \in \{1, \ldots, m\} \) is in the base of a semiretract \( S \).

Now we present a construction of an automaton which recognizes the base of a semiretract \( S \). For any \( L \in \{L_{\text{init}}, L_1, \ldots, L_m\} \) let us consider the language \( \{left(k)\mid k \in L\} \). If a word \( w \) is a prefix of a word from \( \{left(k)\mid k \in L\} \) then \( w \) defines a state \( q_w \). We denote the set of all states obtained in this way by \( Q(L) \). There is an edge \( (q_{w_1}, a, q_{w_2}) \) between states \( q_{w_1}, q_{w_2} \in Q(L) \) if and only if \( w_1a = w_2 \). We denote the set of all edges obtained in this way by \( E(L) \). Note that for any \( k \in L \) there exists a path labeled by \( left(k) \) from \( q_1 \) (the state for the empty word) to \( q_{left(k)} \).

For any \( R \in \{R_{\text{final}}, R_1, \ldots, R_m\} \) let us consider the language \( \{right(k)\mid k \in R\} \). If the word \( w \) is a suffix of a word from \( \{right(k)\mid k \in R\} \) then \( w \) defines a state \( q_w \). We denote the set of all states obtained in this way by \( Q(R) \). There is an edge \( (q_{w_1}, a, q_{w_2}) \) between states \( q_{w_1}, q_{w_2} \in Q(R) \) if and only if \( w_1a = w_2 \). We denote the set of all edges obtained in this way by \( E(R) \). Note that for any \( k \in R \) there exists a path labeled by \( right(k) \) from \( q_{right(k)} \) to \( q_1 \) (the state for the empty word).

Let \( x \in \{1, \ldots, m\} \). Suppose now that all states in \( Q(L_x) \) and \( Q(R_x) \) are distinguishable. If it is necessary we write an upper index \( R_x \) or \( L_x \) to underline that a state is in \( Q(R_x) \) or \( Q(L_x) \). Assume that \( sepx = a_1 \ldots a_k \) where \( a_1, \ldots, a_k \in A \). We define a set
\[ Q(R_x, L_x) = Q(R_x) \cup Q(L_x) \cup \{q_1, \ldots, q_{k-1}\} \]
and we suppose that the sets involved in this sum are pairwise disjoint. Then we define the set of edges \( E(R_x, L_x) \) putting
\[ E(R_x, L_x) = E(R_x) \cup E(L_x) \cup \{(q_{R_k}, w_1, q_1), (q_1, w_2, q_2), \ldots, (q_{k-1}, w_k, q_{L_k})\}. \]

Note that, by the construction, for any \( k_1 \in R_x, k_2 \in L_x \) there exists a path labeled by \( right(k_1)sepx left(k_2) \) from \( q_{right(k_1)} \) to \( q_{left(k_2)} \).

Finally, assume that all states in \( Q(L_{\text{init}}) \), \( Q(R_1, L_1), \ldots, Q(R_m, L_m) \), \( Q(R_{\text{final}}) \) are distinguishable. We define an automaton \( A_S = (Q_S, E_S, I_S, T_S) \) recognizing the base of a semiretract \( S \) as follows. The set of states is equal to \( Q_S = Q(L_{\text{init}}) \cup Q(R_{\text{final}}) \cup \bigcup_{j=1}^{m} Q(R_j, L_j) \). Let \( k \in K \) and assume that \( k \in R_x \) and \( k \in L_y \) for some \( x \in \{1, \ldots, m, \text{final}\}, y \in \{\text{init}, 1, \ldots, m\} \). Then there exist two states \( q_{left(k)} \in Q(L_y) \) and \( q_{right(k)} \in Q(R_x) \). We connect these states with an edge \( (q_{left(k)}, k, q_{right(k)}) \). We repeat that procedure for every key \( k \in K \) and we denote by \( E_1 \) the set of all edges obtained in this way. We put \( E_S = E_1 \cup E(L_{\text{init}}) \cup E(R_{\text{final}}) \cup \bigcup_{j=1}^{m} E(R_j, L_j) \). Finally, we put \( q_1 \in Q(L_{\text{init}}) \) as the only initial state and \( q_1 \in Q(R_{\text{final}}) \) as the only final state.

By the construction of the automaton \( A_S = (Q_S, E_S, I_S, T_S) \) and by Facts 5.2 and 5.3 we have the following statement.
Lemma 5.4. The automaton \( A_S = (Q_S, E_S, I_S, T_S) \) described above is minimal, deterministic and recognizes the base of a semiretract \( S \).

Proof. By the construction, the automaton \( A \) is deterministic. It is not hard to verify that the sets of all words \( L(q) \) for any \( q \in \tilde{Q} \) are pairwise different, where \( L(q) \) denotes the set of all words that occurs as a label on a path from \( q \) to the terminal state. Hence, the automaton is minimal. \( \square \)

It is possible to propose a data structure that allows us to construct an automaton \( A_S \) for a semiretract \( S = \bigcap_{i=1}^n C_i^* \), \( C_1, \ldots, C_n \subset A^* \) key-codes in time

\[
O(\max(n, \log |A|) \ast (|C_1| + \cdots + |C_n|)).
\]

References