A New Interpretation of the Decipherability

János Falucskai
College of Nyíregyháza
falu@nyf.hu

2011.07.21

We define the "quasi code" H as follows: Let Σ and Δ be two finite alphabets. Denote H a finite subset of $2^{\Delta^+} \setminus \emptyset$. We define the function $\bar{f} : \Sigma \rightarrow H$, where $\bar{f}$ is called "quasi coding" of Σ. A quasi code H is called decipherable if, whenever $f(x_1), \ldots, f(x_n), f(y_1), \ldots, f(y_m)$ are in H and satisfy $f(x_1) \ldots f(x_n) = f(y_1) \ldots f(y_m)$, then $n = m$ and $f(x_i) = f(y_i)$ for all i, $1 \leq i \leq n$.

Example:

$$\Sigma = \{a, b\}$$

$$\Delta = \{1, 0\}$$

$$2^{\Delta^+} \setminus \emptyset = \{\{0\}, \{1\}, \{0, 1\}, \{00\}, \{01\} \ldots\}$$

$$H = \{\{0, 1\}, \{0, 110, 1\}\}$$

$$a \longrightarrow \{0, 1\}$$

$$b \longrightarrow \{0, 110, 1\}$$

In general (non quasi codes) a code is a set of sequences of letters:
$C = \{01, 0, 100\}$

$a \longrightarrow 01$

$b \longrightarrow 0$

$c \longrightarrow 100$

The main question: decipherability
$0100 = 0 \quad 100 = 01 \quad 0 \quad 0$

$0100 = bc = abb$

The decipherability only depends on the code set for quasi codes, too.

Basic notions

We call the set Σ an *alphabet,* the elements of Σ *letters*. A *word* over Σ is a finite sequence of elements of some finite non-empty set Σ. The *empty word* λ consisting of zero letters. The *length* $|w|$ of a word w is the number of letters in w. Thus $|\lambda| = 0$. If $u = x_1 \cdots x_k$ and $v = x_{k+1} \cdots x_\ell$ are words over an alphabet Σ (with $x_1, \ldots, x_k, x_{k+1}, \ldots, x_\ell \in \Sigma$) then their *catenation* (which is also called their *product*) $uv = x_1 \cdots x_k x_{k+1} \cdots x_\ell$ is also a word over Σ. In addition, for every word $u = x_1 \cdots x_k$ over Σ (with $x_1, \ldots, x_k \in \Sigma$), $u\lambda = \lambda u = u$ ($= x_1 \cdots x_k$). Moreover, $\lambda\lambda = \lambda$. Obviously, for every $u, v \in \Sigma^*$, $|uv| = |u| + |v|$. Clearly, then, for all words u, v, w (over Σ) $u(vw) = (uv)w$. Catenation is an associative operation and the empty word λ is the identity with respect to catenation. We extend this operation on words to sets.

Let U, V be sets of words.

Then the *catenation* (or *product*) of these two sets is $UV = \{uv : u \in U, v \in V\}$.

A word u is a *factor* (or *subword*) of a word v if $v = v_1 u v_2$ for some words v_1 and v_2.

If v_1 is empty we say that u is a *prefix* of v, and if v_2 is empty, we say that u is a *suffix* of v. Since catenation is an associative operation, for arbitrary $X_1, \ldots, X_n \subseteq \Sigma^*$ the set $X = X_1 \cdots X_n$ is uniquely defined. We say that $X_1 \cdots X_n$ is a *decomposition* of X.

$\{abc, abac\} = \{a\}\{bc, bac\} = \{ab\}\{c, ac\}$

If u is a subword (prefix, suffix) of v such that $u \neq v$ then we speak about *proper subword* (*proper prefix, proper suffix*).

If the nonempty set $X \subseteq \Sigma^*$ is closed under taking factors of its elements, then X is called a *factorial set*.

$\{abc, \lambda, a, b, c, ab, bc\}$

By definition, the empty set is not factorial, and each factorial set contains at least the empty word λ.

Similarly, if a nonempty set $X \subseteq \Sigma^*$ contains all non-empty prefixes of its elements (i.e., is closed under taking a non-empty prefix), we say that it is *prefixial.*

$\{abc, a, ab\}$

Analogously, if it is closed under taking a non-empty suffix, we say that it is *suffixial.*

$\{abc, c, bc\}$

Clearly, each factorial set is prefixial and also suffixial.

Proposition 1 *Every catenation of (finitely many) prefixial (suffixial) sets is also prefixial (suffixial).*

Proof 1 *Given two nonempty prefixial (suffixial) sets $X_1, X_2 \subseteq \Sigma^*$, let $u \in X_1$ and $v \in X_2$. It is enough to prove that all prefixes (suffixes) of uv are in X_1X_2. Let r be a prefix (suffix) of uv with $|u| \leq |r|$ ($|v| \leq |r|$). Then there exists a decomposition $r = ur'$ ($r = r'v$), where r' is a prefix of v (suffix of u). But X_1 is prefixial (suffixial). Thus $r' \in X_2$ ($r' \in X_1$). Hence $r \in X_1X_2$. Suppose that r is a prefix (suffix) of uv with $|u| > |r|$ ($|v| > |r|$). Then r is a a prefix of u (suffix of v) and thus $r \in X_1$ ($r \in X_2$). Because λ is a prefix (suffix) of all words and $X_2 \neq \emptyset$ ($X_1 \neq \emptyset$), $\lambda \in X_2$ ($\lambda \in X_1$). Hence $r \in X_1X_2$ again.* $\square$

Remark 1 *Every catenation of (finitely many) factorial sets is also factorial.*

(Avgustinovich, S., Frid, A.: A unique decomposition theorem for factorial languages (2005))

Proposition 2 *Given a finite nonempty set $X \subset \Sigma^*$, for all nonempty suffixial sets $X_1, X_2 \subseteq \Sigma^*$, $XX_1 = XX_2$ implies $X_1 = X_2$.*

Proof 2 *Suppose that, contrary to our statement, there are a finite nonempty set $X \subset \Sigma^*$, nonempty suffixial sets $X_1, X_2 \subseteq \Sigma^*$ having $XX_1 = XX_2$ and $X_1 \neq X_2$. In this case one of X_1 and X_2 should have an element which is not in the another one. Say, $r \in X_2$ but $r \notin X_1$.*

Because of the finiteness of X, there exists a word $u \in X$ which is not a proper prefix of any word in X. (For example, u has this property if it is one *of the longest words in X.) By $XX_1 = XX_2$ and $ur \in XX_2$, there are words $u_1 \in X, r_1 \in X_1$ with $u_1 r_1 = ur$.*

By our conditions, u is not a proper prefix of u_1. Therefore, $u_1 r_1 = ur$ implies that r is a suffix of r_1. Recall that X_1 is suffixial. Therefore, contrary to our assumptions, $r_1 \in X_1$ implies $r \in X_1$ because r is a suffix of r_1. This completes the proof. $\square$

Remark 2 *The above statement can not be extended for arbitrary infinite sets $X \subseteq \Sigma^*$. Take, for example, $X_1 = \{x_1\}, X_2 = \{x_2\}, x_1, x_2 \in \Sigma, x_1 \neq x_2$ and let $X \subseteq \Sigma^*$ be an arbitrary infinite set having the property that*

$$\forall r \in X : rx_1, rx_2 \in X.$$

Then $XX_1 = XX_2$, but $X_1 \neq X_2$.

A factorial set $X \subseteq \Sigma^*$ is said to be *indecomposable* if $X = AB$ implies $X = A$ or $X = B$ for all factorial sets $A, B \subseteq \Sigma^*$; otherwise we say that X is *decomposable.* Given set $X \subseteq \Sigma^*$, a collection of indecomposable factorial languages $X_1, \ldots, X_n \subseteq \Sigma^*$, we say that $X = X_1 \cdots X_n$ is a *canonical decomposition* of X if one of the following two cases arises:

- $X = X_1 = \{\lambda\}, n = 1$;
- $X \neq \{\lambda\}, X_i \neq \{\lambda\}, i \in \{1, \ldots, n\}$, moreover, $X \neq X_1 \cdots X_{i-1} X_i' X_{i+1} \cdots X_n$ for each $i \in \{1, \ldots, n\}$ and a factorial language $X_i' \subsetneqq X_i$.

Theorem 1 *Each factorial set X has a unique canonical decomposition into factorial sets.*

(Avgustinovich, S., Frid, A.: A unique decomposition theorem for factorial languages (2005))

Quasi codes

For every $x_i \in \Sigma$ we define the set of strings H_i by $H_i = \{p_{i_1}, \ldots p_{i_m}\} \in 2^{\Delta^+} \setminus \emptyset$. Let us interpret the decipherability on the mapping $f(x_i) = H_i$.

A *quasi code* H over Δ is a finite subset of $2^{\Delta^+} \setminus \emptyset$. The elements of a quasi code H are called *code sets*, the elements of H^* are called *messages*. Let the injective mapping $f : \Sigma^+ \to H$ be given. Let the equation

$$f(x_1 \ldots x_n) = f(x_1) \ldots f(x_n); \forall x_i \in \Sigma$$

hold, therefore f can be given by the function $\bar{f}$, where

$$\bar{f} : \Sigma \to H.$$

The function $\bar{f} : \Sigma \to H$ is the *determination of quasi code H belonging to* Σ. The function $f : \Sigma^+ \to H$ is called *quasi coding*.

Let the decipherability of quasi codes be defined analogously as in the case of verbatim codes, i.e. the mapping is decipherable if from the equation

$$f(x_1)\dots f(x_n) = f(y_1)\dots f(y_m)$$

we get, that

$$n = m \text{ and } f(x_i) = f(y_i),\ x_i = y_i.$$

We say that a quasi code H is decipherable, if every message has at most one decomposition. Formally, if the equation

$$f(x_1)\dots f(x_n) = f(y_1)\dots f(y_m)$$

holds, then $n = m$ and $f(x_1) = f(y_1), \dots, f(x_n) = f(y_n)$.

By Remark 1, every catenation of factorial sets is also factorial. Therefore, using Theorem 1, we can derive the following result.

Corollary 3 *Every quasi code* $X_1, \ldots, X_n$, *with* $X_i \nsubseteqq X_j, i \neq j, 1 \leq i, j \leq n$, *consisting of indecomposable factorial sets is uniquely decipherable.*

Criteria of decipherability of quasi codes

Let a set $A \subseteq \Sigma^*$ is called prefix-free for a set $B \subseteq \Sigma^*$, if $\exists\, a \in A$ such that $a\alpha \neq b$ and $b\alpha \neq a$ for $\forall \alpha \in \Sigma^*$ and for $\forall b \in B$. That is $\exists\, a \in A$ such that a is not a prefix of any $b \in B$ and there is no $b \in B$ such that b is a prefix of a.

Example 1 *Let $A = \{baa, ab, b\}$ and let $B = \{baa, bba\}$. In this case the set A is prefix-free for the set B, because $ab \in A$ is not a prefix of any element of the set B and for any element of B the element is not a prefix of ab.*

Example 2 *Let $A = \{baa, ab, b\} \quad B = \{baa, a, bba, aa\}$. In this case the set A is not prefix-free for the set B. The string $ab \in A$ is not a prefix of any element of the set B, but $a \in B$ is a prefix of $ab \in A$.*

Proposition 4 *The properties of the relation* prefix-free for a set*:*

- *the relation is irreflexive*
- *the relation is not symmetric*
- *the relation is not transitive*

Two sets $A, B \subseteq \Sigma^*$ are prefix-free (for each other), if A is prefix free for B, or B is prefix-free for A. Here "or" does not mean "exclusive or".

Example 3 *The following sets A_i, B_i are prefix-free for each other:*
$A_1 = \{ab\}, B_1 = \{a, aa\}; A_2 = \{a, aa\}, B_2 = \{ab\}; A_3 = \{a\}, B_3 = \{b\}$

Proposition 5 *The properties of the relation prefix-free for each other:*

- *the relation is irreflexive*
- *the relation is symmetric*
- *the relation is not transitive*

Let the set H consist of subsets of Σ^*. The set H is called *prefix-free,* if any two elements of H are prefix-free for each other.

Theorem 2 *If a quasi code consisting of nonempty suffixial sets is prefix-free, then the quasi code is decipherable.*

Proof 3 *The proof we give here is an indirect one. Assume that a quasi code consisting of nonempty sufficial sets is prefix-free, but it is not decipherable. Since the quasi code is not decipherable, there is a set G, such that we get G from the quasi code in at least two ways. Denote by H_i the set $f(x_i)$. Take the following two different decompositions:*

$$G = H_{i_1} \dots H_{i_s} \text{ and } G = H_{j_1} \dots H_{j_t}$$

Because of the indirect hypothesis there is a positive integer l such that $H_{i_k} = H_{j_k}$ for $\forall k < l$. But, $H_{i_l} \neq H_{j_l}$. If $l = 1$ then $H_{i_1} \neq H_{j_1}$ and $H_{i_1} \dots H_{i_s} = H_{j_1} \dots H_{j_t}$.

Otherwise, using the suffix Proposition 1, all of the decompositions $H_{i_1}\dots H_{i_{l-1}}$, $H_{j_1}\dots H_{j_{l-1}}, H_{i_l}\dots H_{i_s}, H_{j_l}\dots H_{j_t}$ *are suffixial sets. Therefore, applying Proposition 2, from equations*

$$H_{i_1}\dots H_{i_{l-1}} H_{i_l}\dots H_{i_s} = H_{j_1}\dots H_{l-1} H_{j_l}\dots H_{j_t}$$

and

$$H_{i_1}\dots H_{i_{l-1}} = H_{j_1}\dots H_{j_{l-1}}$$

we have that

$$G' = H_{i_l}\dots H_{i_s} = H_{j_l}\dots H_{j_t}.$$

Moreover, $H_{i_l} \neq H_{j_l}$ *is assumed.*

Thus, $\forall p \in G'$ *could be written in the form* $p = x\beta = y\gamma$*, where* $x \in H_{i_l}$*,* $y \in H_{j_l}$*. That is,* $x\alpha = y$ *or* $x = y\alpha$*, where* $\alpha \in \Sigma^*$*.*

It is easy to see that there exists $p \in G'$ *such that* $p = x\beta = y\gamma$ *for* $\forall x \in H_{i_l}$ *and for* $\forall y \in H_{j_l}$ *because of the catenation property of sets. Therefore, there is* $\alpha \in \Sigma^*$ *for all* $x \in H_{i_l}$ *such that* $x\alpha = y$ *or* $x = y\alpha$ *holds for some* $y \in H_{j_l}$*.*

Consequently, H_{i_l} *is not prefix-free for* H_{j_l} *(analogously, we have that* H_{j_l} *is not prefix-free for* H_{i_l}*). Thus, the sets* H_{i_l} *and* H_{j_l} *are not prefix-free for each other. Therefore, the quasi code is not prefix-free. We have a contradiction and hence the theorem is proved.* $\square$

Theorem 3 *There exists a quasi code consisting of nonempty prefixial sets such that it is prefix-free but not decipherable.*

Proof 4 *Let* $H_1 = \{b, ba, baa\}$, $H_2 = \{a, aa, aaa, aaaa, ab, aaab, aaaab\}$, $H_3 = \{a, aa, aaa, aaaa, ab, aab, aaaab\}$. *None of the elements of* $H_2 \cup H_3$ *is a prefix of some element in* H_1 *and none of the elements of* H_1 *is a prefix of some element in* $H_2 \cup H_3$. *On the other hand,* $aaab \in H_2$ *is not a prefix of any element of* H_3 *and* $aab \in H_3$ *is not a prefix of any element in* H_2. *Therefore, the quasi code* H_1, H_2, H_3 *is prefix-free. On the other hand, it is clear that all of* H_1, H_2, H_3 *are prefixial. To show* $H_1H_2 = H_1H_3$, *we have to consider the catenations of all elements in* H_1 *and* $aaab \in H_2$, *moreover, the catenations of all elements in* H_1 *and* $aab \in H_3$. *But* $(b)(aaab) = (ba)(aab), (ba)(aaab) = (b)(aaaab), (baa)(aaab) = (ba)(aaaab)$, *and simultaneously,* $(b)(aab) = (ba)(ab), (ba)(aab) = (b)(aaab), (baa)(aab) = (b)(aaaab)$. *Therefore,* $H_1H_2 = H_1H_3$ *holds* *such that $H_2 \neq H_3$. In other words, the considered quasi code is not decipherable.* $\square$

Remark 3 *It is easy to see that there are decipherable prefix quasi codes. One of the most simple examples is $H_1 = \{a\}, H_2 = \{ab\}$.*

Theorem 4 *If A and A^k are elements of a quasi code H, then the quasi code H is not decipherable.*

Proof 5 *Let $f(x) = A$; $f(y) = A^k$. Thus, $\underbrace{f(x)...f(x)}_{k} = \underbrace{A \ldots A}_{k} = A^k$. $f(y) = A^k$. Therefore,*

$$\exists n \neq m : f(x_{i_1})...f(x_{i_n}) = f(x_{j_1})...f(x_{j_m}).$$

Consequently the quasi code is not decipherable. $\square$

We give a generalized form of the previous theorem:

Theorem 5 *If* $\exists\, A = \prod_{i=1}^{m} A_i^{k_i} \in H$ $(m > 1, k_i \geq 1, A_1, A_2, \ldots, A_m \in H)$, *then the quasi code* H *is not decipherable.*

Proof 6 *Let* $f(x) = A = \prod_{i=1}^{m} A_i^{k_i}; f(y_1) = A_1, \ldots f(y_m) = A_m$. *This implies that* $f(x) = A = \prod_{i=1}^{m} A_i^{k_i}$ *and* $\underbrace{f(y_1)\cdots f(y_1)}_{k_1} \ldots \ldots \underbrace{f(y_m)\cdots f(y_m)}_{k_m} = \prod_{i=1}^{m} A_i^{k_i}$. *Thus,*

$$\exists n \neq m : f(x_{i_1})...f(x_{i_n}) = f(x_{j_1})...f(x_{j_m}).$$

Therefore, the quasi code H *is not decipherable.* $\square$

Application of the Sardinas–Patterson algorithm for quasi codes

The decipherability of codes was solved by the Sardinas–Patterson algorithm. Let us try to use it for quasi codes. The application of the algorithm forms the following power set system:

Let X and Y be two subsets of the set $2^{\Delta^+} \setminus \emptyset$. Let $X^{-1}Y$ denote the following set: $\{C \mid \exists A \in X, B \in Y : AC = B\}$.

As a straightforward extension of the Sardinas-Patterson algorithm, consider the following algorithm (called Quasi-Code SP):

Let the set H be a subset of the set $2^{\Delta^+} \setminus \emptyset$, and

$$\begin{aligned} U_1 = & \; H^{-1}H \setminus \{\lambda\} \\ U_2 = & \; H^{-1}U_1 \cup U_1^{-1}H \\ \vdots & \\ U_{n+1} = & \; H^{-1}U_n \cup U_n^{-1}H. \end{aligned} \tag{1}$$

If there exist $i > j \geq 1$ with $U_i = U_j$ and $\lambda \notin U_k$ for any $k < i$ then let the Quasi-Code SP algorithm answer that the quasi-code is decipherable. Otherwise let it answer that the quasi-code is not decipherable.

Theorem 6 *There exist quasi-codes for which the Quasi-Code SP-algorithm does not give a correct answer.*

Proof 7 *Based on the Sardinas–Patterson theory our conjecture was the following:*
If $\exists i, j$ *such that* $U_i = U_j$ *and* $\{\lambda\} \notin U_i$*, then the quasi code* H *is decipherable. Unfortunately, this statement is false. The behaviour of sets of strings is not similar to the behaviour of strings with respect to the operation of catenation. The following holds for strings:*

Let $x, y, z \in \Sigma^* \setminus \{\lambda\}$*, then* $xy = xz$ *implies that* $y = z$*. The Sardinas–Patterson algorithm is based on this connection. Of course, each set* X *admits two* trivial decompositions $X = AB$*, where one of the sets* A *and* B *is equal to* $\{\lambda\}$*, where* λ *is the empty word, and the other is equal to set* X*.*

If a set has only trivial decompositions, it is natural to call it a prime set*. However, even a finite set can have several non-trivial decompositions to prime sets, and an infinite set can have none of them . Our conjecture was the following: if the sets* A, B, C *are prime sets, then* $AB = AC$ *implies* $B = C$*. It is not true, for example in the sets* L, L_1, L_2 *are prime sets, but* $LL_1 = LL_2$ *holds. Namely*

$$L = \{b, ba, baa, c, caa, caaa, caaaa\}$$

$$L_1 = \{ab, aaab, aaaab, c\}, L_2 = \{ab, aaaab, c\}$$

Thus, if we form a quasi code with these sets, that is $H = \{L, L_1, L_2\}$ *and if we apply the Sardinas–Patterson-like algorithm for* H*, then we have* $U_1 = \emptyset$ *by the first step. (Note that* $U_1 = \emptyset$ *implies* $U_2 = \emptyset$*, i.e.,* $U_i = U_j$ *with* $i = 1$ *and* $j = 2$*.) The quasi code seems decipherable according to the Sardinas–Patterson-like algorithm, but, in fact, it is not. Let the following* *quasi code*

$$f(x_1) = L, f(x_2) = L_1, f(x_3) = L_2$$

be given. $f(x_2) \neq f(x_3)$*, but the equation* $f(x_1)f(x_2) = f(x_1)f(x_3)$ *holds. Therefore, the quasi code is not decipherable.* $\square$

By our explanation, it seems that there exists no straightforward extension of the Sardinas-Patterson algorithm for quasi-codes.

Decomposition of quasi codes

We use the algorithm which decides the prime property of sets to determine a decomposition (Mateescu, A., Salomaa, A., Yu, S.: On the decomposition of finite languages):

Let R be a regular language over the alphabet Σ, and let $\mathcal{A} = (Q, \Sigma, \delta, q_0, Q_F)$ be the minimal deterministic finite automaton for R. (Here Q is the set of states, q_0 is the initial state, Q_F is the set of final states, and δ is the transition function.) We extend δ to words over Σ. Thus, $\delta(q, w) = q'$ means that the word w takes A from the state q to the state q' (and, by definition, $R = \{w \in \Sigma^* \mid \delta(q_0, w) \in QF\}$). For a nonempty subset $P \subseteq Q$, we consider the following two languages:

$$R_1^P = \{w | \delta(q_0, w) \in P\},$$

$$R_2^P = \{w | \delta(p, w) \in Q_F, p \in P\}.$$

Theorem 7 *Let R and $\mathcal{A}$ be defined as above. Assume that $R = L_1 L_2$, where L_1 and L_2 are arbitrary languages. Define $P \subseteq Q$ by*

$$P = \{p \in Q | \delta(q_0, w) = p, \text{ for some } w \in L_1\}.$$

Then $R = R_1^P R_2^P$, moreover, $L_1 \subseteq R_1^P$ and $L_2 \subseteq R_2^P$.

By definition, a nonempty subset $P \subseteq Q$ is a decomposition set (for a regular language R), if $R = R_1^P R_2^P$. The decomposition $R = R_1^P R_2^P$ is referred to as the decomposition of R induced by the decomposition set P. We say that the decomposition $L = L_1 L_2$ of a language L is included in the decomposition
$L = L_1' L_2'$ if $L_i \subseteq L_i', i = 1, 2$.

Theorem 8 *Every decomposition of a regular language R is included in a decomposition of R induced by a decomposition set. The problem of primality is decidable for regular languages.*

Using these notations we form the following automaton:

Let $\mathcal{A} = (Q, \Delta, \delta, q_0, Q_F)$ be the minimal deterministic finite automaton for some finite set $X \subset H^*$ where $H = \{f(x_1), \ldots, f(x_n)\}$ is a given quasi code. (Here Q is the set of states, q_0 is the initial state, Q_F is the set of final states, and δ is the transition function.) We extend δ to words over Σ as we did above. Thus, $\delta(x, w) = y$ means that the word w takes A from the state x to the state y (and, by definition, $X = \{w \in \Delta^* \mid \delta(q_0, w) \in Q_F\}$). For non-empty subsets $P_1, P_2 \subseteq Q$, we consider the following language:

$$R_{P_1,P_2} = \{w|\delta(p, w) \in P_2, p \in P_1\}$$

Theorem 9 *Let H and $\mathcal{A}$ be defined as above. Assume, that $X = f(x_{i_1}) \cdots \cdots f(x_{i_k})$, where $f(x_{i_1}), \ldots f(x_{i_k}) \in H$. Define the sets $P_0, \ldots, P_k \subseteq Q$ by*

$$P_0 = \{q_0\}$$

$$P_1 = \{p \in Q | \delta(q_0, w) = p, \text{ for some } w \in f(x_{i_1})\}.$$

(It is evident, that $f(x_{i_1}) \subseteq R_{\{q_0\}, P_1}$.)

$$P_2 = \{p \in Q | \delta(q_0, w) = p, \text{ for some } w \in f(x_{i_1}) f(x_{i_2})\}.$$

$$\vdots$$

$$P_k = Q_F = \{p \in Q | \delta(q_0, w) = p, \text{ for some } w \in f(x_{i_1}) \cdots f(x_{i_k})\}.$$

Then $X = R_{P_0, P_1} \cdots R_{P_{k-1}, P_k}$ and $f(x_{i_1}) \subseteq R_{P_0, P_1}, \ldots, f(x_{i_k}) \subseteq R_{P_{k-1}, P_k}$.

Proof 8 *First, we establish the inclusions. To prove the inclusion $f(x_{i_m}) \subseteq R_{P_{m-1},P_m}$, assume the contrary: for some $w_{i_m} \in f(x_{i_m})$ and $p \in P_{m-1}$, $\delta(p, w_{i_m}) \notin P_m$. Choose a word $w \in f(x_{i_1}) \cdots f(x_{i_{m-1}})$ such that $\delta(q_0, w) = p$. Since $ww_{i_m} \in f(x_{i_1}) \cdots f(x_{i_m})$, we have $\delta(q_0, ww_{i_m}) \in P_m$. But $\delta(q_0, ww_{i_m}) = \delta(p, w_{i_m}) \notin P_m$. This contradiction proves the inclusion $f(x_{i_m}) \subseteq R_{P_{m-1},P_m}$.*

Second, we establish the statement $X = R_{P_0,P_1} \cdots R_{P_{k-1},P_k}$. Consider an arbitrary word $w_1 \cdots w_k$, where $w_m \in R_{P_{m-1},P_m}$. Since, $w_1 \in R_{P_0,P_1}$, $\delta(q_0, w_1) = p_1$. By the definition of P_1, we have $p_1 \in P_1$. By the definition of R_{P_1,P_2}, $\delta(p_1, w_2) = p_2 \in P_2$ and similarly for all $1 \leq m \leq k$ that (by the definition of R_{P_{m-1},P_m}), $\delta(p_{m-1}, w_m) = p_m \in P_m$. Thus, $\delta(q_0, w_1 \cdots w_k) = \delta(p_1, w_2 \cdots w_k) = \cdots = \delta(p_{k-1}, w_k) = p_k \in Q_F$, and thus, $w_1 \cdots w_k \in X$, therefore $X \supseteq R_{P_0,P_1} \cdots R_{P_{k-1},P_k}$. Consider an arbitrary $w \in X$. We can write $w =$

$w_1 \cdots w_k$, *where* $w_m \in f(x_{i_m})$, $1 \leq m \leq k$. *By the already proved inclusion* $f(x_{i_m}) \subseteq R_{P_{m-1},P_m}$, *we conclude that* $w_m \in R_{P_{m-1},P_m}$, *where* $1 \leq m \leq k$. *Thus,* $w = w_1 \cdots w_k \in R_{P_0,P_1} \cdots R_{P_{k-1},P_k}$. *Therefore,* $X \subseteq R_{P_0,P_1} \cdots R_{P_{k-1},P_k}$. *Having these two inclusions, we get that* $X = R_{P_0,P_1} \cdots R_{P_{k-1},P_k}$. $\square$

Consider a set of nonempty subsets $\{P_1, \ldots, P_{k-1}\}$, where $P_m \subseteq Q$, $m \in \{1, \ldots, k-1\}$ is a decomposition set for a finite set X and a quasi code
$H = \{f(x_1), \ldots, f(x_n)\}$, where $X = R_{\{q_0\},P_1} \cdots R_{P_{k-1},Q_F}$. The decomposition
$X = R_{\{q_0\},P_1} \cdots R_{P_{k-1},Q_F}$ will be referred to as the *decomposition of X induced by the decomposition set* $\{P_1, \ldots, P_{k-1}\}$. We say that the decomposition $X = X_1 \cdots X_k$ of a finite set X is *included in the decomposition* $X = X'_1 \cdots X'_k$ if $X_m \subseteq X'_m$, $m = 1, 2, \ldots k$.

Theorem 10 *Every decomposition of a finite set X is included in a decomposition of X induced by a decomposition set. The problem of decipherability is decidable for finite sets.*

Proof 9 *The first part of Theorem 10 follows by Theorem 9. To perform the verification for all possible decompositions of a finite set X, check through all sets of nonempty subsets $\{P_1, \ldots P_{k-1}\}$, where $P_m \subseteq Q$. If more than one of them induces a nontrivial decomposition, we conclude that H is not decipherable.* $\square$

Of course, there are non-decipherable quasi codes such that they have one or zero decompositions for a set. For example, the set $H = \Big\{\{a\}, \{aa\}, \{b\}\Big\}$ has one decomposition $\{b\}\{a\}$ for the set $X = \{ba\}$.