

Egyetemi doktori (PhD) értekezés tézisei

KÓDOK FELBONTHATÓSÁGA

Falucskai János

Témavezető: Dr. Fazekas Gábor



DEBRECENI EGYETEM
Matematika- és Számítástudományok Doktori Iskola
Debrecen, 2013.

Tartalomjegyzék – Contents

1	A doktori értekezés előzményei és célkitűzései	1
2	Az értekezés új tudományos eredményei	4
2.1	Egy, a felbonthatóságot eldöntő új algoritmus . . .	4
2.2	A felbonthatóság egy új értelmezése	7
2.3	Kvázikódok nemfelbonthatóságának egy szükséges feltétele	8
1	Introduction, research objectives	10
2	Results	13
2.1	A new algorithm for the decipherability	13
2.2	A new interpretation of the decipherability	15
2.3	A necessary condition	17
	Hivatkozások–References	18
	Publikációs lista–List of papers	22
	Előadások–List of talks	24
	Egyéb–Others	25

1. A doktori értekezés előzményei és célkitűzései

A kódelmélet az információelmélet egyik legfontosabb alkalmazási területe. Kialakulásának időpontját ezért elsősorban az információelmülethez szokás kötni, ami az 1940-es években Shannon munkásságával vette kezdetét.

Az információelmélettel együtt megszületett kódelmélet alapvetően három területtel foglalkozik:

- Egyértelmű dekódolhatóság
- Optimalizálás, vagy más néven adattömörítés
- Hibafelismerés és hibajavítás

Jelen értekezés a kódok felbonthatóságával foglalkozik, gyakran nevezik a felbontható kódokat egyértelműen megfejthető (*uniquely decipherable*), vagy röviden *UD* kódoknak. A felbonthatóság problémáját az teszi érdekessé, hogy a hogy a kód optimalizálás megoldása változó hosszúságú kódokra épül.

A változó hosszúságú kódokkal mélyebben először *Schützenberger* [Sch55], valamint *Gilbert és Moore* [GM59], majd később *Shyr* [Shy01] foglalkozott. *Schützenberger* vizsgálatai nyomán került kapcsolatba a kódelmélet a nemkommutatív algebra elméletével, melynek korai eredményei *Nivat* [Niv66] foglalta össze.

A változó hosszúságú kódok és az automaták kapcsolatának alapos összefoglalóját találhatjuk *Berstel és Perrin* [BP85] könyvében, melynek folyamatosan frissített verziója szabadon elérhető a világhálón [BPR09]. Az automaták elmélete széles

körben vizsgált és jól kidolgozott matematikai elmélet, világos útmutatást adnak a rájuk épülő implementációk kivitelezéséhez.

A felbonthatóság problémájára először *Sardinas* és *Patterson* [SP53] adott megoldást, ezt általában *Sardinas–Patterson* algoritmusként ismert. A felbonthatóság kérdésével ezek után többen is foglalkoztak, pl. *Bandyopadhyay* [Ban63], *Levenshtein* [Lev64], *Riley* [Ril67], *de Luca* [dL76]. A felbonthatóságot eldöntő algoritmusok optimalizálásával kapcsolatban *Spehner* [Spe76], *Rodeh* [Rod82], *Apostolico* és *Giancarlo* [AG84], *McCloskey* [McC96] végzett kutatásokat. A felbonthatóságot eldöntő algoritmus pontos bonyolultsága jelenleg még nem ismert, de a kérdéssel kapcsolatban *Hoffmann* [Hof84], *Galil* [Gal85] és *McCloskey* [McC96] valamint *König* [Kön94] értékes eredményeket értek el.

Az automatákat alkalmazó algoritmusokra fogunk részletesebben kitérni. A kódok egyértelmű megfeleltetősége az automataelméletben is jelenlévő reguláris kifejezések többértelműségi problémájának a speciális esete, melyre vannak eldöntő algoritmusok, melyek megtalálhatóak például *Eilenberg* [Eil74], vagy *Aho* és társai [AHU75] munkáiban. A kódok és a reguláris kifejezések kapcsolatára *Brzozowsky* [Brz67] mutatott rá. A kódok felbonthatóságának eldöntésére több automataelméleti alapú algoritmus is ismert, az ilyen megközelítésnek már vannak hagyományai, mint például a „virág” automata [BP85], az automaták összefűzött szorzata [Kön94] és a Tsuji-féle konstrukció [Tsu01].

Alapfogalmak

Az A ábécé egy véges nem üres halmaz, elemei betűk, vagy más néven szimbólumok. Az A ábécé feletti w szó az A ábécé véges számú elemeiből képzett szimbólumsorozat.

$$w = (a_1, a_2, \dots, a_n), \quad a_i \in A.$$

Az A ábécé feletti teljes nyelv az A feletti összes szó által alkotott halmaz, jelölése A^* . Az üres jelsorozatot *üresszónak* nevezzük és λ -val jelöljük. Az üres jelsorozatról könnyű belátni, hogy a konkatenációra nézve az egységelem szerepét tölti be. Az A feletti C kód az A^+ véges részhalmaza. A C kód elemeit *kódszavaknak*, a C^* elemeit pedig *üzeneteknek* nevezzük. Egy C kódot *UD (uniquely decipherable)* kódnak, azaz felbonthatónak nevezünk, ha minden egyes üzenetnek egyértelmű felbontása van, tehát ha teljesül az $x_1x_2 \cdots x_n = y_1y_2 \cdots y_m$ egyenlőség, ahol $x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_m \in C$, akkor $n = m$ és $x_1 = y_1, \dots, x_n = y_n$.

Automata alatt egy véges automatát, azaz egy $\mathcal{A} = (Q, I, Q_F, A, \delta)$ ötöst értünk, ahol a Q egy nem üres és véges halmaz, $I \subseteq Q$ a kezdő állapotok halmaza, $Q_F \subseteq Q$ az elfogadó állapotok halmaza, A egy nem üres és véges halmaz a bemenő jelek halmaza és

$$\delta = \{(p, q, a) | p, q \in Q, a \in A\} \subseteq Q^2 \times A$$

a mozgási szabályok halmaza.

Legyen X és Y két részhalmaza az A^+ halmaznak, valamint legyen $x \in X$ és $y \in Y$. Jelölje $X^{-1}Y$ a következő halmazt:

w egy eleme a $X^{-1}Y$ halmaznak, ha $\exists x \in X, y \in Y$ úgy, hogy $xw = y$. Amennyiben $X = \{x\}$, $Y = \{y\}$, azaz X és Y egyelemű halmazok, akkor $x^{-1}y$ jelöli azt a w szót, melyre $xw = y$. Legyen a C halmaz a A^+ halmaz részhalmaza, és legyen

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

1. Tétel (Sardinas – Patterson, [BP85]). *A $C \subset A^+$ halmaz felbontható kód akkor, és csak akkor, ha a fentebb definiált U_n halmazok közül egyetlenegy sem tartalmazza az üres szót.*

2. Az értekezés új tudományos eredményei

2.1. Egy, a felbonthatóságot eldöntő új algoritmus

Ha a w_i kódszó $x_1x_2 \dots x_n$, $x_j \in A$ alakú, akkor a hozzátartozó $\mathcal{A}(\{w_i\})$ automata $\mathcal{A}(\{w_i\}) = (Q^{(i)}, q_\lambda, Q_F^{(i)}, A, \delta^{(i)})$ legyen, ahol $Q^{(i)}$ az állapotok halmaza, q_λ az $\mathcal{A}(\{w_i\})$ kezdőállapota és $Q_F^{(i)}$ az elfogadó állapotot tartalmazó egy elemű halmaz. Egyesítsük a kódszavakhoz tartozó automatákat, ekkor az $\mathcal{A}(\{w_1, \dots, w_n\})$ automatát kapjuk a $C = \{w_1, \dots, w_n\}$ kód

esetén. Jelölje $x_1^{(l)}$ az l . kódszó első szimbólumát. Tehát

$$\mathcal{A}(C) = (q, Q_F = \{1, \dots, n\}, Q = Q^{(1)} \cup \dots \cup Q^{(n)}, A, \delta),$$

ahol

$$\delta = \delta^{(1)} \cup \dots \cup \delta^{(n)} \cup \{\delta(1, x_1^{(1)}) = q_{x_1^{(1)}}, \dots, \delta(1, x_1^{(n)}) = q_{x_1^{(n)}}, \dots,$$

$$\delta(n, x_1^{(1)}) = q_{x_1^{(1)}}, \dots, \delta(n, x_1^{(n)}) = q_{x_1^{(n)}}\}.$$

A konstrukcióból következik, hogy az $\mathcal{A}(C)$ automata pontosan a C^+ nyelvet fogja elfogadni.

Ha a v jelsorozat előállítható a C kód segítségével, akkor az $\mathcal{A}(C)$ automata elfogadja a v jelsorozatot, azaz $\mathcal{A}(C)$ elolvassa és valamely elfogadó állapotban áll meg. Ha v több, mint egyféleképpen előállhat C kódszavai konkatenációjaként, akkor különböző állapotsekvenciákat kapunk a különböző elolvasások során. Ezeket a különböző utakat egyesítjük a determinisztikus automata által.

2. Tétel ([Fal04, Fal06, Fal11b]). *Egy kód felbontható akkor és csak akkor, ha az $\mathcal{A}_D(C)$ automatában legfeljebb egy – az $\mathcal{A}(C)$ automatához tartozó – elfogadó állapot jelenik meg bármely szabály jobboldalán, azaz bármely szabályra igaz az, hogy ha $\delta(\{q_{i_1}, \dots, q_{i_n}\}, x) = \{q_{j_1}, \dots, q_{j_m}\}$ az $\mathcal{A}_D(C)$ automatában, akkor nem létezik $l \neq k$ úgy, hogy $q_{j_l} \in Q_F^{\mathcal{A}(C)}$ és $q_{j_k} \in Q_F^{\mathcal{A}(C)}$ is egyszerre teljesül.*

A Sardinas–Patterson féle algoritmus a többszörös felbontásnál jelentkező maradványok vizsgálatán alapszik. A többszörös

felbontásnak szükséges feltétele, hogy különböző kódszavakkal kezdődjenek a felbontások, tehát a kód ne legyen prefixmentes. Valamely kódszó tehát valódi kezdőszelete valamely másik kódszónak, emiatt lesz nemdeterminisztikus az $\mathcal{A}(C)$ automata. Egy $q \in \mathcal{A}(C)$ állapot által indukált maradványokat megkapjuk, ha az $\mathcal{A}(C)$ automatában megkeressük azokat a jelsorozatokat, melyekkel a q állapotból – csak elutasító állapotokat érintve – elfogadó állapotba kerülünk.

1. Következmény ([Fal08a]). Minden α maradvány eleme az $\mathcal{A}_D(C)$ automata valamely állapotában valamely $\mathcal{N}(w)$ állapottal együtt szereplő q állapot által indukált maradvány halmaznak, ahol $\alpha \in U$, $\mathcal{N}(w)$ és q eleme a $Q^{\mathcal{A}(C)}$ halmaznak.

2. Következmény ([Fal08a]). Ha az $\mathcal{A}_D(C)$ automata egy q_D állapotában valamely $\mathcal{N}(w)$ állapottal együtt szerepel egy q állapot, akkor a q által indukált maradvány halmaz minden α eleme benne van az U halmazban, ahol $\alpha \in U$, $\mathcal{N}(w)$ és q eleme a $Q^{\mathcal{A}(C)}$ halmaznak.

Jelölje $R(q)$ az $\mathcal{A}(C)$ automata q állapota által indukált maradványok halmazát, és legyenek az U_i halmazok a Sardinas–Patterson algoritmus alapján létrehozott halmazok. Ekkor az 1. és a 2. következményekből adódik a következő állítás.

3. Következmény ([Fal07, Fal08a]).

$$\bigcup_{i \geq 1} U_i = \bigcup_{\{\mathcal{N}(w), q\} \subseteq q_D} R(q),$$

ahol $\mathcal{N}(w) \in Q_F^{\mathcal{A}(C)}$, $q_D \in Q_F^{\mathcal{A}_D(C)}$

2.2. A felbonthatóság egy új értelmezése

Értelmezzük a felbonthatóságot egy olyan leképezésen, ahol egy $x \in \Sigma$ szimbólumhoz nem csak egy, hanem tetszőleges véges számú

$$\{p_1, \dots, p_m\} \in 2^{\Delta^*} \setminus \emptyset$$

korlátos jelsorozat tartozzon, azaz $f(x) = \{p_1, \dots, p_m\}$. A Δ feletti H kvázikód a $2^{\Delta^*} \setminus \emptyset$ véges részhalmaza. A H kvázikód elemeit *kódhalmazoknak*, a H^+ elemeit pedig *üzeneteknek* nevezük. Egy $\bar{f} : \Sigma \rightarrow H$ bijektív függvényt a Σ halmazhoz tartozó H kvázikód megadásának nevezzük. Tetszőleges $\bar{f} : \Sigma \rightarrow H$ kvázikód megadása esetén az $f(x_1 \cdots x_n) = \bar{f}(x_1) \cdots \bar{f}(x_n)$, $x_1, \dots, x_n \in \Sigma$ feltételnek eleget tevő $f : \Sigma^+ \rightarrow H^+$ függvényt kvázikódolásnak nevezzük.

A felbonthatóságot a szokásos módon értelmezem, tehát az $f : \Sigma^+ \rightarrow H^+$ kvázikódolás felbontható, ha az

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

egyenlőségből következik, hogy

$$n = m \text{ és } f(x_i) = f(y_i) \text{ (} x_i = y_i \text{)}$$

Egy H kvázikódot *UD (uniquely decipherable)* kvázikódnak, azaz felbonthatónak nevezzük, ha minden egyes üzenetnek egyértelmű felbontása van, tehát ha teljesül az

$$f(x_1) \cdots f(x_n) = f(y_1) \cdots f(y_m),$$

egyenlőség, ahol $f(x_1), \dots, f(x_n), f(y_1), \dots, f(y_m) \in H$, akkor $n = m$ és $f(x_1) = f(y_1), \dots, f(x_n) = f(y_n)$.

3. Tétel ([Fal11a]). *Ha egy nem üres, a $\{\lambda\}$ halmazt nem tartalmazó suffixiál halmazokból álló kvázikód λ -prefixmentes, akkor a kvázikód felbontható.*

2.3. Kvázikódok nemfelbonthatóságának egy szükséges feltétele

1. Definíció ([Fal11a]). *Egy determinisztikus automatát jelöltnek nevezünk, ha az állapotainak jelölései tartalmazzák a nemdeterminisztikus automata generáló állapotainak jelölését. A generátor állapotok közül a végállapotokat az utolsó kódszóval indexeljük. Egy állapot jelölthalmazán a generátor állapotainak halmazát értjük.*

4. Tétel ([Fal11a]). *Egy kvázikód nemfelbonthatóságának egy szükséges feltétele, hogy a hozzátartozó jelölt determinisztikus automatában létezzen legalább egy olyan állapot, mely jelölthalmazában szerepel az eredeti nemdeterminisztikus automata legalább két végállapota.*

2. Definíció ([Fal11a]). *Legyenek adva az A, B, C_1, \dots, C_m halmazok. Azt mondjuk, hogy az A és a B halmaz kölcsönösen párosítva van a C_1, \dots, C_m halmazokban, ha az összes $a \in A$ esetén létezik legalább egy $b \in B$ és $1 \leq i \leq m$ úgy, hogy $C_i \supseteq \{a, b\}$ és viszont, azaz az összes $b \in B$ esetén létezik legalább egy $a \in A$ és $1 \leq i \leq m$ úgy, hogy $C_i \supseteq \{a, b\}$.*

Az $f(x) = \{w_1, \dots, w_n\}$ esetén jelölje $\tilde{f}(x)$ az $\{x_{w_1}, \dots, x_{w_n}\}$ halmazt.

5. Tétel ([Fal08b, Fal11a]). *Ha egy $H = \{f(x_1), \dots, f(x_n)\}$ kvázikód nemfelbontható, akkor a H kvázikódhoz tartozó jelölt determinisztikus automatában léteznek a $C_1 \dots, C_m$ állapotok és az x_i, x_j generátor állapotok úgy, hogy $\tilde{f}(x_i)$ és $\tilde{f}(x_j)$ kölcsönösen párosítva van a $C_1 \dots, C_m$ állapotokhoz tartozó jelölthalmazokban.*

1 Introduction, research objectives

The code theory is one of the most important application area of the information theory. The beginning of code theory coincides to beginning of information theory. The information theory started by the work of Shannon in the late 1940's.

The areas of code theory:

- Uniquely decipherability
- Data compressing
- Error detection and correction

This dissertation is on the decipherability of codes. The codes are frequently called decipherable codes or uniquely decipherable codes or just codes. The problem of decipherability is made interesting by the variable length codes, which are used for the solution of code optimization.

The variable-length codes were investigated foremost in depth by *Schützenberger* [Sch55], and also by *Gilbert and Moore* [GM59], and later by *Shyr* [Shy01]. The direction followed by *Schützenberger* consists in linking the theory of codes with classical noncommutative algebra. An early account of it can be found in the article of *Nivat* [Niv66].

The automata theory is widely investigated and elaborated mathematical theory. We have correct instructions for the constructions of the implementations based on automata theory.

A review of the connection of the variable-length codes and automaton is presented in the book of *Berstel* and *Per-*

rin [BP85]. It can be found on the Internet for free download [BPR09].

A test for the uniquely decipherability for codes is given by *Sardinas* and *Patterson* [SP53] foremost. It is known as *Sardinas–Patterson* algorithm. The result involved a number of papers by , *Markov* [Mar62], *Bandyopadhyay* [Ban63], *Levenshtein* [Lev64], *Riley* [Ril67], *de Luca* [dL76]. The design of an efficient algorithm is described by *Spehner* [Spe76], *Rodeh* [Rod82], *Apostolico* and *Giancarlo* [AG84], *McCloskey* [McC96]. The complexity of the algorithm of the deciding the decipherability is not known, but *Hoffmann* [Hof84], *Galil* [Gal85] and *McCloskey* [McC96] and *König* [Kön94] have reached useful results.

We are investigated the algorithms based on automata theory. The uniquely decipherability of the codes is a special case of a problem in the automata theory, namely testing whether a given rational expression is unambiguous. Standard decision procedures exist for this question, see *Eilenberg* [Eil74], or *Aho* et al. The connection between codes and regular expressions has been pointed out by *Brzozowsky* [Brz67].

There are several algorithms for deciding the decipherability. Some algorithms based on automata theory, and use automata, these approaches have antecedents. For example, the "flower" automaton [BP85], latticed products of automata [Kön94] and the Tsuji like construction [Tsu01].

Preliminaries

The *alphabet* A is a finite nonempty set. The *word* w over the alphabet A is a finite sequence of symbols of the alphabet A .

$$w = (a_1, a_2, \dots, a_n), \quad a_i \in A.$$

The *empty word* λ is the empty string consisting of zero symbols. The *Kleene star of A* denoted by A^* is the set of all strings of finite length consisting of symbols in A including the empty word λ . It is easy to see, that the empty word λ is the identity of alphabet A for the operation of concatenation. The *code* C over the alphabet A is a finite subset of A^+ . The elements of the code C are the *code words*, and the elements of C^* are the *messages*. A code C is *UD (uniquely decipherable)* code, if every message has at most one factorization with respect to the code C , that is if $x_1x_2 \cdots x_n = y_1y_2 \cdots y_m$ holds, where $x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_m \in C$, then $n = m$ and $x_1 = y_1, \dots, x_n = y_n$.

By an automaton we mean a *finite automaton*, that is an $\mathcal{A} = (Q, I, Q_F, A, \delta)$ 5-tuple. The set Q of states is a nonempty finite set, the set $I \subseteq Q$ is the set of start states, the set $Q_F \subseteq Q$ is the set of accept states, the nonempty finite set A is the input alphabet, and the set

$$\delta = \{(p, q, a) | p, q \in Q, a \in A\} \subseteq Q^2 \times A$$

is the set of transition rules.

Let the sets X and Y two subsets of the set A^+ , and let $x \in X$ and $y \in Y$. We define the set $X^{-1}Y$ by the following way:

the string w is an element of the set $X^{-1}Y$, if $\exists x \in X, y \in Y$ such that, $xw = y$. If $X = \{x\}$, $Y = \{y\}$, that is, the sets X and Y are singletons, then denote $x^{-1}y$ the word w , such that $xw = y$. Let the set C a subset of the set A^+ , and let

$$\begin{aligned} U_1 &= C^{-1}C \setminus \{\lambda\} \\ U_2 &= C^{-1}U_1 \cup U_1^{-1}C \\ &\vdots \\ U_n &= C^{-1}U_{n-1} \cup U_{n-1}^{-1}C \end{aligned}$$

Theorem 1 (Sardinas – Patterson, [BP85]) *The set $C \subset A^+$ is a decipherable code if, and only if none of U_n contains the empty word.*

2 Results

2.1 A new algorithm for deciding the decipherability

If the code word w_i has the form $x_1x_2 \dots x_n$, $x_j \in A$, then let be the automaton $\mathcal{A}(\{w_i\}) = (Q^{(i)}, q_\lambda, Q_F^{(i)}, A, \delta^{(i)})$ concerning w_i , where the set $Q^{(i)}$ is the set of states, the set q_λ is the start state of the automaton $\mathcal{A}(\{w_i\})$ and the set $Q_F^{(i)}$ is the singleton of the one accept state. Let us combine the automata of code words. Thus, we get the automaton $\mathcal{A}(\{w_1, \dots, w_n\})$ for the code $C = \{w_1, \dots, w_n\}$. Denote $x_1^{(l)}$ the first symbol of w_l . Therefore,

$$\mathcal{A}(C) = (q, Q_F = \{1, \dots, n\}, Q = Q^{(1)} \cup \dots \cup Q^{(n)}, A, \delta),$$

where

$$\delta = \delta^{(1)} \cup \dots \cup \delta^{(n)} \cup \{ \delta(1, x_1^{(1)}) = q_{x_1^{(1)}}, \dots, \delta(1, x_1^{(n)}) = q_{x_1^{(n)}}, \dots, \\ \delta(n, x_1^{(1)}) = q_{x_1^{(1)}}, \dots, \delta(n, x_1^{(n)}) = q_{x_1^{(n)}} \}.$$

Thus, the automaton $\mathcal{A}(C)$ accepts the language C^+ .

Therefore, if the string v can be produced by code C , then the automaton $\mathcal{A}(C)$ accepts v , that is the automaton $\mathcal{A}(C)$ reads v and stays in an accept state. If v can be produced by multiple ways by concatenation of codewords of C , then we get different sequences of states by the different readings. These different ways are combined into one way in the deterministic automaton.

Theorem 2 ([Fal04, Fal06, Fal11b]) *A code is decipherable if, and only if in the automaton $\mathcal{A}_D(C)$ at most one – being in the automaton $\mathcal{A}(C)$ – accept state appears on the right side of any transaction rule. That is, for any transaction rule the following holds: if the transaction rule $\delta(\{q_{i_1}, \dots, q_{i_n}\}, x) = \{q_{j_1}, \dots, q_{j_m}\}$ is in the automaton $\mathcal{A}_D(C)$, then don't exist $l \neq k$ such that, $q_{j_l} \in Q_F^{A(C)}$ and $q_{j_k} \in Q_F^{A(C)}$ hold.*

The Sardinas – Patterson algorithm is based on the investigation of multiple reminders. The factorizations must begin with different code words. This is a necessary condition of the multiple factorization. Therefore, the code is not prefix-free. Thus, there is a code word, which is perfect prefix part of some other code word. Consequently, the automaton $\mathcal{A}(C)$ is non-deterministic. We get the reminders induced by a state $q \in \mathcal{A}(C)$, if we collect

the read strings from state q to an accept state in the automaton $\mathcal{A}(C)$. We touch only reject states during the way.

Corollary 1 ([Fal08a]) *Every reminder α is an element of a reminder set induced by q , where in the automaton $\mathcal{A}_D(C)$ the state q is together with a state $\mathcal{N}(w)$. $\alpha \in U$, $\mathcal{N}(w)$ and q are elements of the set $Q^{\mathcal{A}(C)}$.*

Corollary 2 ([Fal08a]) *If a state $\mathcal{N}(w)$ and a state q are together in a state $q_D \in \mathcal{A}_D(C)$, then every element α of the reminder set induced by q is in the set U , where $\alpha \in U$, $\mathcal{N}(w)$ and q are elements of the set $Q^{\mathcal{A}(C)}$.*

Denote $R(q)$ the reminder set induced by $q \in \mathcal{A}(C)$ and let the sets U_i related to Sardinas–Patterson algorithm. Then from Corollary 1. and Corollary 2. we get the following statement.

Corollary 3 ([Fal07, Fal08a])

$$\bigcup_{i \geq 1} U_i = \bigcup_{\{\mathcal{N}(w), q\} \subseteq q_D} R(q),$$

where $\mathcal{N}(w) \in Q_F^{\mathcal{A}(C)}$, $q_D \in Q_F^{\mathcal{A}_D(C)}$.

2.2 A new interpretation of the decipherability

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

A *quasi code* H over Δ is a finite subset of the set $2^{\Delta^*} \setminus \emptyset$. The elements of a quasi code H are called *code sets*, the elements of H^+ are called *messages*.

The function $\bar{f} : \Sigma \rightarrow H$ is said the *determination of quasi code* H belongs to Σ . Let the equation

$$f(x_1 \cdots x_n) = \bar{f}(x_1) \cdots \bar{f}(x_n); \forall x_i \in \Sigma$$

holds. The function $f : \Sigma^+ \rightarrow H^+$ is called *quasi coding*. Let the decipherability of quasi codes be defined analogously as in the case of verbatim code, 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

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

Definition 1 ([Fal08b, Fal11a]) *A quasi code H is called UD (uniquely decipherable) quasi code, 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 $f(x_1) = f(y_1), \dots, f(x_n) = f(y_n)$.

Theorem 3 ([Fal11a]) *Every non-empty λ -prefix-free quasi-code consist of suffixial sets -which does not contain λ set- is decipherable.*

2.3 A necessary condition of the not-decipherability

Definition 2 ([Fal11a]) *A deterministic automaton is called a marked one, if the notations of states of automaton consist of the notations of the generator states of the non-deterministic automaton. The final states of the generator states are subscripted by the last codeword. The marked set of a state is the set of its generator states.*

Theorem 4 ([Fal11a]) *If a quasi code is not decipherable, then there is a state in the marked deterministic automaton such that its marked set contains at least two final states of the original non-deterministic automaton.*

Definition 3 ([Fal11a]) *Let A, B, C_1, \dots, C_m be sets. A and B are called mutually paired for C_1, \dots, C_m , if there is at least one $b \in B$ and there is $1 \leq i \leq m$ such that $C_i \supseteq \{a, b\}$ for all $a \in A$, and conversely, that is there is at least one $a \in A$ and there is $1 \leq i \leq m$ such that $C_i \supseteq \{a, b\}$ for all $b \in B$.*

Denote $\tilde{f}(x)$ the set $\{x_{w_1}, \dots, x_{w_n}\}$, where $f(x) = \{w_1, \dots, w_n\}$.

Theorem 5 ([Fal08b, Fal11a]) *If a quasi code $H = \{f(x_1), \dots, f(x_n)\}$ is not decipherable, then there are C_1, \dots, C_m marked states and x_i, x_j in the deterministic automaton of H such that $\tilde{f}(x_i)$ and $\tilde{f}(x_j)$ are mutually paired for marked states of C_1, \dots, C_m .*

We note that to find sufficient condition for the non-decipherability is open problem.

Hivatkozások–References

- [AG84] A. Apostolico and R. Giancarlo. Pattern matching machine implementation of a fast test for unique decipherability. *Inform. Process. Lett.*, 18(3):155–158, 1984.
- [AHU75] Alfred V. Aho, John E. Hopcroft, and Jeffrey D. Ullman. *The design and analysis of computer algorithms*. Addison-Wesley Publishing Co., Reading, Mass.-London-Amsterdam, 1975. Second printing, Addison-Wesley Series in Computer Science and Information Processing.
- [Ban63] G. Bandyopadhyay. A simple proof of the decipherability criterion of Sardinas and Patterson. *Information and Control*, 6:331–336, 1963.
- [BP85] J. Berstel and D. Perrin. *Theory of codes*, volume 117 of *Pure and Applied Mathematics*. Academic Press Inc., Orlando, FL, 1985.
- [BPR09] J. Berstel, D. Perrin, and C. Reutenauer. *Codes and Automata*. Cambridge University Press, 2009. <http://www-igm.univ-mlv.fr/~berstel/LivreCodes/CodesAutomata.pdf>.
- [Brz67] J. A. Brzozowski. Roots of star events. *J. Assoc. Comput. Mach.*, 14:466–477, 1967.
- [dL76] Aldo de Luca. A note on variable length codes. *Information and Control*, 32(3):263–271, 1976.

- [Eil74] Samuel Eilenberg. *Automata, languages, and machines. Vol. A*. Academic Press [A subsidiary of Harcourt Brace Jovanovich, Publishers], New York, 1974. Pure and Applied Mathematics, Vol. 58.
- [Fal04] J. Falucskai. DFA of non unique decodable code. In *6th International Conference on Applied Informatics*, pages 303–309, Eger, 2004.
- [Fal06] J. Falucskai. On a test for codes. *Acta Mathematica Academiae Paedagogicae Nyíregyháziensis*, 22:121–125, 2006.
- [Fal07] J. Falucskai. Some algorithms concerning uniquely decipherable codes. In *7th International Conference on Applied Informatics*, volume 2, pages 229–236, Eger, 2007.
- [Fal08a] J. Falucskai. On equivalence of two tests for codes. *Acta Mathematica Academiae Paedagogicae Nyíregyháziensis*, 24:249–256, 2008.
- [Fal08b] J. Falucskai. Változó hosszúságú kódok. In *II. Nyíregyházi Doktorandusz (PhD/DLA) Konferencia*, pages 219–222, Nyíregyháza, 2008.
- [Fal11a] J. Falucskai. A new interpretation of the decipherability. Közlésre benyújtva, *Acta Informatica*, 2011.
- [Fal11b] János Falucskai. A novel test for unique decipherability of codes. *Publ. Math. Debrecen*, 78(3-4):535–541, 2011.

- [Gal85] Zvi Galil. Open problems in stringology. In *Combinatorial algorithms on words (Maratea, 1984)*, volume 12 of *NATO Adv. Sci. Inst. Ser. F Comput. Systems Sci.*, pages 1–8. Springer, Berlin, 1985.
- [GM59] E. N. Gilbert and E. F. Moore. Variable-length binary encodings. *Bell System Tech. J.*, 38:933–967, 1959.
- [Hof84] Christoph M. Hoffmann. A note on unique decipherability. In *Mathematical foundations of computer science, 1984 (Prague, 1984)*, volume 176 of *Lecture Notes in Comput. Sci.*, pages 50–63. Springer, Berlin, 1984.
- [Kön94] R. König. Lectures on codes. Internal Reports of the IMMD I, Friedrich Alexander University of Erlangen-Nürnberg, 1994.
- [Lev64] V. I. Levenšteĭn. Some properties of coding and self-adjusting automata for decoding messages. *Problemy Kibernet. No.*, 11:63–121, 1964.
- [Mar62] Al. A. Markov. On alphabet coding. *Soviet. Phys. Dokl.*, 6:553–554, 1962.
- [McC96] Robert McCloskey. An $O(n^2)$ time algorithm for deciding whether a regular language is a code. In *Proceedings of the 8th International Conference of Computing and Information, ICCI'96 (Waterloo, ON)*, volume 2, pages 79–89 (electronic), 1996.

- [Niv66] Maurice Nivat. Éléments de la théorie générale des codes. In *Automata Theory*, pages 278–294. Academic Press, New York, 1966.
- [Ril67] John A. Riley. The Sardinas-Patterson and Levenshtein theorems. *Information and Control*, 10:120–136, 1967.
- [Rod82] M. Rodeh. A fast test for unique decipherability based on suffix trees. *IEEE Trans. Inform. Theory*, IT-28:648–651, 1982.
- [Sch55] M. Schützenberger. Une théorie algébrique du codage. *Séminaire Dubreil-Pisot 1955-56*, page Exposé 15, 1955.
- [Shy01] H. J. Shyr. *Free Monoids and Languages*. Hon Min Book Company, Taichung, Taiwan, 2001.
- [SP53] A. A. Sardinas and C. W. Patterson. A necessary and sufficient condition for the unique decomposition of coded messages. *IRE Internat. Conv. Rec.*, 8:104, 1953.
- [Spe76] J. C. Spehner. *Quelques problèmes d’extension, de conjugaison et de présentation des sous-monoïdes d’un monoïde libre*. PhD thesis, Université Paris 7, 1976.
- [Tsu01] Kayoko Tsuji. An automaton for deciding whether a given set of words is a code. *RIMS Kokyuroku*, 1222:123–127, 2001.

Publikációs lista–List of papers

- [1] J. Falucskai. A novel test for unique decipherability of codes. *Publicationes Mathematicae Debrecen*, volume 78, Fasc.: 3–4, 535–541, 2011.
DOI: 10.5486/PMD.2011.4716
- [2] J. Falucskai. On the k -reversibility of finite automata. *Annales Mathematicae et Informaticae*, 36:71–75, 2009.
ZENTRALBLATT Zbl 1212.68088
- [3] J. Falucskai. On equivalence of two tests for codes. *Acta Mathematica Academiae Paedagogicae Nyíregyháziensis*, 24:249–256, 2008.
ZENTRALBLATT Zbl 1164.94303
- [4] J. Falucskai. Some algorithms concerning uniquely decipherable codes. In *7th International Conference on Applied Informatics*, volume 2, pages 229–236, Eger, 2007.
ZENTRALBLATT Zbl 1179.94075
- [5] J. Falucskai. On a test for codes. *Acta Mathematica Academiae Paedagogicae Nyíregyháziensis*, 22:121–125, 2006.
ZENTRALBLATT Zbl 1120.94013
- [6] J. Falucskai. DFA of non unique decodable code. In *6th International Conference on Applied Informatics*, volume 1, pages 303–309, Eger, 2004.
ZENTRALBLATT Zbl 1074.68507

-
- [7] J. Falucskai. An application of regular expressions. In *microCAD 2003 International Scientific Conference*, volume Section N: Applied Information Engineering, pages 59–64, Miskolc, 2003.
- [8] J. Falucskai. Unique decodability of codes. In *microCAD 2002 International Scientific Conference*, volume Section H: Applied Information Engineering, pages 77–82, Miskolc, 2002.

Előadások–List of talks

- [1] J. Falucskai. A New Interpretation of the Decipherability. *Brno University of Technology*, 2011.07.21
- [2] J. Falucskai. Some Algorithms Concerning Uniquely Decipherable Codes. *Brno University of Technology*, 2011.03.29
- [3] J. Falucskai. Változó hosszúságú kódok. In *II. Nyíregyházi Doktorandusz (PhD/DLA) Konferencia*, pages 219–222, Nyíregyháza, 2008.
- [4] J. Falucskai. Kódok felbonthatóságát vizsgáló program. *Magyar Tudomány Napja*, Nyíregyháza, 2004.
- [5] J. Falucskai. Reguláris kifejezések alkalmazása. *Az MTA Sz-Sz-B Megyei Tudományos Testület tudományos ülése*, Nyíregyháza, 2003.
- [6] J. Falucskai. Reguláris kódok alkalmazása. *Magyar Tudomány Napja*, Nyíregyháza, 2003.
- [7] J. Falucskai. Nemprefix kódok automatái. *Az MTA Sz-Sz-B Megyei Tudományos Testület tudományos ülése*, Nyíregyháza, 2002.
- [8] J. Falucskai. Reguláris kifejezések. *Magyar Tudomány Napja*, Nyíregyháza, 2002.
- [9] J. Falucskai. Reguláris halmaz egyenletrendszerek és nem felbontható kódok. *Magyar Tudomány Napja*, Nyíregyháza, 2001.

- [10] J. Falucskai. Kódok és automaták összefüggéseinek néhány vizsgálata. *Magyar Tudomány Napja*, Nyíregyháza, 2000.
- [11] J. Falucskai. Véges mondattal rendelkező nyelvek és a kód-elmélet néhány összefüggése. *Az MTA Sz-Sz-B Megyei Tudományos Testület tudományos ülése*, Nyíregyháza, 2000.

Egyéb–Others

- [1] J. Falucskai. A new interpretation of the decipherability. Submitted for publication, *Acta Informatica*, 2011.
- [2] J. Falucskai. A számítástudomány néhány analóg rendszere. *Természettudományi közlemények, Nyíregyházi Főiskola*, 4:89–99, Nyíregyháza, 2004.
- [3] Á. Kuki, J. Falucskai, and K. Tarnay. *Bevezetés a formális nyelvek és automaták alkalmazásába*, volume 7 of *Az MTA Sz-Sz-B Megyei Tudományos Testületének Közleményei*. Nyíregyháza, 1993.