Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

# Dodatočná informácia a triedy jazykov

Bakalárska práca

2018

Jozef Martiš

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

# Dodatočná informácia a triedy jazykov

Bakalárska práca

| | |
|---|---|
| Študijný program: | Informatika |
| Študijný odbor: | 2508 Informatika |
| Školiace pracovisko: | Katedra informatiky |
| Školiteľ: | prof. RNDr. Branislav Rovan, PhD. |

Bratislava, 2018
Jozef Martiš

49659180

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

# ZADANIE ZÁVEREČNEJ PRÁCE

| | |
|---|---|
| **Meno a priezvisko študenta:** | Jozef Martiš |
| **Študijný program:** | informatika (Jednoodborové štúdium, bakalársky I. st., denná forma) |
| **Študijný odbor:** | informatika |
| **Typ záverečnej práce:** | bakalárska |
| **Jazyk záverečnej práce:** | slovenský |
| **Sekundárny jazyk:** | anglický |

**Názov:** Dodatočná informácia a triedy jazykov
*Auxiliary information and families of languages*

**Anotácia:** Práca je ďalším príspevkom k porozumeniu pojmu užitočnosť informácie. Skúma vlastnosti tried jazykov, ktoré môžeme priradiť k danej informácii na základe užitočnosti pre ich akceptovanie konečnými automatmi.

**Cieľ:** Nadviazať na doterajší výskum pojmu informácia. Využiť formalizmus konečných automatov a skúmať vlastnosti tried jazykov, pre akceptovanie ktorých je daná informácia užitočná.

| | |
|---|---|
| **Vedúci:** | prof. RNDr. Branislav Rovan, PhD. |
| **Katedra:** | FMFI.KI - Katedra informatiky |
| **Vedúci katedry:** | prof. RNDr. Martin Škoviera, PhD. |
| **Dátum zadania:** | 29.10.2017 |

**Dátum schválenia:** 30.10.2017

doc. RNDr. Daniel Olejár, PhD.
garant študijného programu

.......................................................... študent

.......................................................... vedúci práce

**Poďakovanie:** Ďakujem vedúcemu tejto bakalárskej práce, prof. RNDr. Branislavovi Rovanovi, PhD., za jeho odborné i pedagogické vedenie a hlavne za jeho nesmierne charakterný prístup.

# Abstrakt

Táto práca je pokračovaním skúmania pojmu užitočnosť informácie. Skúmame vlastnosti triedy problémov (jazykov), ktorá je definovaná pevne danou dodatočnou informáciou. Ukážeme, že trieda problémov, ktorých riešenie sa touto dodatočnou informáciou zjednoduší, nie je uzavretá na bežné operácie na jazykoch. Skúmame tiež "informačnú silu" jazykov tak, že skúmame, či a ako sa vzťah dvoch daných dodatočných informácií (jazykov) premietne do vzťahu nimi definovaných tried problémov (jazykov).



**Kľúčové slová:** deterministický konečný automat, dodatočná informácia, trieda jazykov, informačná sila

# Abstract

This work is a continuation of the study of auxiliary information. We study families of problems (languages), which are defined by a given auxiliary information. We show that the family of problems, whose solution is simplified by this auxiliary information, is not closed under common operations on languages. We also study "information power" of languages so that we examine whether and how the relationship of two given auxiliary information (languages) will be projected into relationship of families of problems (languages) defined by them.


**Keywords:** deterministic finite automaton, auxiliary information, family of languages, information power

# Obsah

**Úvod** **1**

**1 Základné pojmy** **2**

**2 Triedy definované dodatočnou informáciou** **6**

**3 Informačná sila jazykov** **12**

**Záver** **16**

# Zoznam obrázkov

2.1 Automat pre jazyk $L_{1_{new}}$ . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9

3.1 Vzťah $\mathscr{L}(L_1) \cap \mathscr{L}(L_2)$ s triedou $\mathscr{L}(L_1 \cap L_2)$ . . . . . . . . . . . . . . . . 14

# Zoznam tabuliek

2.1 Uzáverové vlastnosti triedy $\mathscr{L}(L_{adv})$ . . . . . . . . . . . . . . . . . . . . . 11

3.1 Porovnanie informačnej sily . . . . . . . . . . . . . . . . . . . . . . . . . . . 15

# Úvod

Táto práca nadväzuje na sériu prác, ktoré začali skúmať niektoré vlastnosti pojmu informácia, ako sú užitočnosť, aktuálnosť a pod. (napr. práce [1], [3], [4]). V doterajších prácach sa skúmala otázka, či (a pre ktoré) problémy (jazyky) existuje užitočná dodatočná informácia, ktorá zjednoduší ich riešenie. V tejto práci zameriame našu pozornosť na triedu jazykov, ktorú tvoria tie jazyky (problémy), ktorým pomôže konkrétna (pevne daná) dodatočná informácia.

V 1. kapitole uvedieme definície deterministického konečného automatu, ďalšie potrebné definície pre túto prácu a nakoniec dodatočnú a užitočnú dodatočnú informáciu.

V 2. kapitole zadefinujeme triedu jazykov užitočnej dodatočnej informácie. Pozrieme sa na správanie tejto triedy tým, že preskúmame jej uzáverové vlastnosti.

V 3. kapitole budeme porovnávať vzťah dodatočných informácií so vzťahom tried jazykov, ktoré sú určené týmito informáciami. Na záver sa pozrieme na triedu jazykov, ktorá je určená prenikom/zjednotením dvoch dodatočných informácií k vzťahu prieniku/zjednotenia dvoch tried, určených týmito dodatočnými informáciami.

# Kapitola 1

# Základné pojmy

V tejto kapitole uvedieme základné pojmy súvisiace s deterministickými konečnými automatmi, s ktorými budeme neskôr pracovať. Celkovo sa budeme zaoberať regulárnymi automatmi. Ďalšie pojmy z teórie formálnych jazykov sa dajú nájsť napr. v [2]. Definujeme aj pojem akceptovania jazyka s danou dodatočnou informáciou.

**Definícia 1.** Deterministický konečný automat *A je pätica (K, $\Sigma$, $\delta$, $q_0$, F), pričom* K *je konečná množina stavov,* $\Sigma$ *je vstupná (konečná) abeceda,* $\mathrm{q}_0 \in K$ *je počiatočný stav automatu,* $\mathrm{F} \subseteq K$ *a* $\delta$*:* $K \times \Sigma \rightarrow K$ *je prechodová funkcia automatu.*

**Poznámka 1.** *Deterministický konečný automat budeme tiež označovať skrátene* DKA

Situáciu v ktorej sa pri spracovaní vstupného slova v danom okamihu nachádza DKA definujeme formálne pojmom *konfigurácia.*

**Definícia 2.** Konfigurácia *DKA A je dvojica (q, u)* $\in K \times \Sigma^*$. *Túto dvojicu interpretujeme tak, že q je momentálny stav automatu A a u predstavuje nespracovanú časť vstupného slova.*

DKA pri spracovaní vstupného slova prechádza v diskrétnych krokoch z jednej konfigurácie do druhej. Formálne definujeme krok výpočtu takto:

**Definícia 3.** Krok výpočtu *DKA A je relácia* $\vdash_A$ *na konfiguráciách definovaná pre* $p, q \in K, a \in \Sigma, w \in \Sigma^*$ *takto:* $(q,\ aw) \vdash_A (p,\ w) \iff p = \delta(q,a)$.

**Poznámka 2.** *Reflexívny a tranzitívny uzáver* $\vdash_A$ *značíme* $\vdash_A^*$

DKA sú prostriedkom pre definovanie jazykov. Konkrétne regulárnych jazykov.

**Definícia 4.** Jazyk *akceptovaný (definovaný) DKA A je množina slov* $L(A) = \{w \mid \exists q \in F; (q_0, w) \vdash_A^* (q, \varepsilon)\}$.

K danému jazyku L môže existovať viac DKA, ktoré ho akceptujú. V tejto práci budeme hľadať čo najjednoduchšie spôsoby definovania jazykov. Ako mieru zložitosti budeme používať počet stavov DKA.

**Definícia 5.** Stavovú zložitosť *regulárneho DKA A, označujeme $\#_s(A)$, definujeme ako počet stavov automatu A, teda $\#_s(A) = |K|$.*

**Definícia 6.** Minimálny DKA pre L *je DKA, ktorý akceptuje L a spomedzi všetkých, ktoré akceptujú L, má najmenší počet stavov. Teda pre minimálny DKA $A_L$ pre L platí, že $L = L(A_L)$ a pre každý DKA $A'$ taký, že $L(A') = L$ platí $\#_s(A') \geq \#_s(A_L)$.*

**Poznámka 3.** *Z Myhill-Nerodovej vety [2] plynie, že pre ľubovoľný regulárny jazyk L minimálny $A_L$ existuje a je jednoznačne daný (až na premenovanie stavov).*

**Definícia 7.** Stavová zložitosť *jazyka L, $sc(L)$, je stavová zložitosť minimálneho DKA pre jazyk L, $sc(L) = \#_s(A_L)$.*

Na regulárny jazyk L a DKA A, ktorý ho akceptuje sa môžeme pozerať aj tak, že L predstavuje nejaký problém, ktorý treba riešiť. Konkrétny prípad toho problému je určiť pre dané slovo $w$ či patrí do L. DKA A potom prestavuje nejaké riešenie tohto problému - dostane na vstup slovo $w$, realizuje na ňom výpočet a slovo akceptuje (alebo nie). DKA $A_L$ potom predstavuje najjednoduchšie možné riešenie problému L.

V tejto práci nadviažeme na práce [1], [3] a [4] ktoré si kládli otázku, či je možné zjednodušiť riešenie problému L, ak budeme o vstupnom slove vedieť nejakú dodatočnú informáciu (radu, advice). Formálnejšie, ak budeme môcť predpokladať, že vstupné slovo patrí do nejakého jazyka $L_{adv}$, môže nám pre definovanie L stačiť jednoduchší automat (nebude musieť overovať všetky vlastnosti, ktoré by $w$ malo spĺňať). Definujeme teraz formálne, čo to znamená, že automat A akceptuje jazyk s radou $L_{adv}$:

**Definícia 8.** *Pre DKA A a jazyk $L_{adv}$ definujeme* jazyk akceptovaný automatom A s dodatočnou informáciou (advice) *takto:*

$$L(A \mid L_{adv}) = L(A) \cap L_{adv}.$$

Keďže nás zaujíma zjednodušovanie riešenia problémov, budeme vyžadovať aby aj $L_{adv}$ bol "jednoduchý". Bude nás zaujímať, či pre regulárny jazyk L daný DKA $A_L$ existujú DKA $A_{new}$ a dodatočná informácia $L_{adv}$, daná DKA $A_{adv}$ také, že $L = L(A_{new}) \cap L_{adv}$ a platí $\#_s(A_{new}) < \#_s(A_L)$ a $\#_s(A_{adv}) < \#_s(A_L)$.

**Definícia 9.** *Jazyk $L_{adv}$ je* užitočná dodatočná informácia *pre jazyk L, ak $sc(L_{adv}) < sc(L)$ a existuje DKA $A_{new}$ [1] taký, že*

- $L = L(A_{new} \mid L_{adv})$
- $\#_s(A_{new}) < sc(L)$.

[1] V ďalších častiach tejto práce budeme tiež namiesto automatu $A_{new}$ používať priamo jazyk $L_{new}$, ktorý tento automat akceptuje.

**Príklad 1.** *Majme jazyk* $L = \{a^{6n} \mid n \in \mathbb{N}\ \}$. *Minimálny automat pre tento jazyk je:* $A = (K, \Sigma, \delta, q_0, F)$, *kde*

- $K = \{q_0, q_1, q_2, q_3, q_4, q_5\}$
- $\Sigma = \{a\}$
- $F = \{q_5\}$
- $\delta$ :

$$\delta(q_0, a) = q_1$$
$$\delta(q_1, a) = q_2$$
$$\delta(q_2, a) = q_3$$
$$\delta(q_3, a) = q_4$$
$$\delta(q_4, a) = q_5$$
$$\delta(q_5, a) = q_0$$

*Minimalitu automatu ľahko vidieť a teda ju nebudeme dokazovať. Užitočnou informáciou pre tento jazyk môže byť jazyk akceptovaný nasledujúcim automatom:* $A_{adv} = (K_{adv}, \Sigma_{adv}, \delta_{adv}, q_{adv0}, F_{adv})$

- $K_{adv} = \{q_{adv0}, q_{adv1}\}$
- $\Sigma_{adv} = \{a\}$
- $F_{adv} = \{q_{adv1}\}$
- $\delta_{adv}$ :

$$\delta_{adv}(q_{adv0}, a) = q_{adv1}$$
$$\delta_{adv}(q_{adv1}, a) = q_{adv0}$$

*Teda* $L(A_{adv})$ *hovorí, že vstupné slovo je párnej dĺžky. Jazyk* $L(A_{adv})$ *je užitočná dodatočná informácia pre jazyk L. Počet stavov* $A_{adv}$ *je zjavne menší ako počet stavov minimálneho automatu A pre jazyk L. Teraz musíme ukázať, že existuje ďalší automat* $A_{new}$ *pre ktorý bude platiť:* $L(A_{new} \mid L_{adv}) = L$, *teda*

$$\#_s(A_{new}) < \#_s(A) = sc(L) = 6.$$

*Zostrojíme teda automat:* $A_{new} = (K_{new}, \Sigma_{new}, \delta_{new}, q_{new_0}, F_{new})$

- $K_{new} = \{q_{new_0}, q_{new_1}, q_{new_2}\}$
- $\Sigma_{new} = \{a\}$

- $F_{new} = \{q_{new_2}\}$
- $\delta_{new}$ :

$$\begin{aligned}
\delta_{new}(q_{new_0}, a) &= q_{new_1} \\
\delta_{new}(q_{new_1}, a) &= q_{new_2} \\
\delta_{new}(q_{new_2}, a) &= q_{new_0}
\end{aligned}$$

*Zrejme*

$$\begin{aligned}
L(A_{new}) &= \{a^{3n} \mid n \in \mathbb{N}\}, \\
L(A_{adv}) &= \{a^{2n} \mid n \in \mathbb{N}\}.
\end{aligned}$$

*Je tiež zrejmé, že platí*

$$\{a^{3n} \mid n \in \mathbb{N}\} \cap \{a^{2n} \mid n \in \mathbb{N}\} = \{a^{6n} \mid n \in \mathbb{N}\}.$$

*Z konštrukcie automatu* $A_{new}$ *vidíme, že* $\#_s(A_{new}) < 6$.

# Kapitola 2

# Triedy definované dodatočnou informáciou

V tejto kapitole k danému jazyku $L_{adv}$ zadefinujeme triedu jazykov, pre ktoré je $L_{adv}$ užitočnou dodatočnou informáciou a budeme skúmať jej uzáverové vlastnosti.

**Definícia 10.** *Nech $L_{adv}$ je jazyk, potom $\mathscr{L}(L_{adv})$ je trieda jazykov definovaná takto: $\mathscr{L}(L_{adv}) = \{L \mid L_{adv}$ je užitočná dodatočná informácia pre $L\}$.*

Predchádzajúca definícia uvádza kľúčový pojem tejto práce. Práve triedy $\mathscr{L}(L_{adv})$, kde $L_{adv}$ predstavuje nejaký regulárny jazyk, sú predmetom nášho skúmania. Najprv preskúmame ich uzáverové vlastnosti. V ďalšej kapitole sa pozrieme na vzťahy medzi triedami v súvislosti so vzťahmi dodatočných informácií (ktoré tieto triedy určujú).

V nasledujúcich tvrdeniach ukážeme, že triedy $\mathscr{L}(L_{adv})$ nemusia byť vo všeobecnosti uzavreté na niektoré operácie.

**Veta 1.** *Existuje $L_{adv} \in \mathscr{R}$ taký, že $\mathscr{L}(L_{adv})$ nie je uzavretá na zreťazenie.*

*Dôkaz.* Uvažujme triedu $\mathscr{L}(A_{adv})$ danú dodatočnou informáciou $L_{adv} = \{a^{3m}, b^{3n} \mid m, n \in \mathbb{N}\}$ a jazyky $\mathrm{L}_1 = \{a^{6m}, b^{12n} \mid m, n \in \mathbb{N}\}$ a $\mathrm{L}_2 = \{a^{12m}, b^{6n} \mid m, n \in \mathbb{N}\}$. Ľahko vidieť, že oba patria do $\mathscr{L}(L_{adv})$. Zreťazením $L_1$ a $L_2$ nám vznikne nový jazyk $\mathrm{L}_{12} = \{a^{6i}a^{12j}, a^{6i}b^{6j}, b^{12i}a^{12j}, b^{12i}b^{6j} \mid i, j \in \mathbb{N}\}$. Majme slovo $w = a^6b^{12}$. Vidno, že $w \in L_{12}$. Aby platilo $L_{12} \in \mathscr{L}(L_{adv})$, musí existovať $A_{new}$ taký, že $L_{12} = L(A_{new} \mid L_{adv})$ a $\#_s(A_{new}) < sc(L_{12})$, teda:

$$L(A_{new} \mid L_{adv}) = L(A_{new}) \cap L_{adv} = L_{12}.$$

Potom by však muselo platiť:

$$w = a^6b^{12} \in L_{adv}.$$

Zjavne $w \notin L_{adv}$, preto $L_1.L_2 \notin \mathscr{L}(L_{adv})$.

□

**Veta 2.** *Existuje* $L_{adv} \in \mathscr{R}$ *taký, že* $\mathscr{L}(L_{adv})$ *nie je uzavretá na iteráciu.*

*Dôkaz.* Podobne ako v prípade zreťazenia. Uvažujeme triedu $\mathscr{L}(L_{adv})$ danú dodatočnou informáciou $\mathrm{L}(A_{adv}) = \{a^{3m}, b^{3n} \mid m, n \in \mathbb{N}\}$. Nech $\mathrm{L} = \{a^{6m}, b^{6n} \mid m, n \in \mathbb{N}\}$. Zjavne $\mathrm{L} \in \mathscr{L}(A_{adv})$. Majme $w$ také, že:

$$w = a^6 b^6 \in L^*.$$

Potom však platí:

$$w \notin L(A_{adv}) \Rightarrow L^* \notin \mathscr{L}(A_{adv}).$$

□

**Veta 3.** *Existuje* $L_{adv} \in \mathscr{R}$ *taký, že* $\mathscr{L}(L_{adv})$ *nie je uzavretá na reverz.*

*Dôkaz.* Majme triedu $\mathscr{L}(L_{adv})$ danú dodatočnou informáciou $L_{adv} = \{a^{3m}b^{3n} \mid m, n \in \mathbb{N}\}$. Ľahko vidieť, že jazyk $\mathrm{L} = \{a^{6m}b^{6n} \mid m, n \in \mathbb{N}\}$ patrí do $\mathscr{L}(L_{adv})$. Aplikujme teraz na jazyk L operáciu reverz. Dostaneme jazyk $\{b^{6m}a^{6n} \mid m, n \in \mathbb{N}\}$. Potom však platí:

$$L^R = \{b^{6m}a^{6n} \mid m, n \in \mathbb{N}\} \ni b^6 a^6 = w \notin L_{adv} \Rightarrow L^R \notin \mathscr{L}(L_{adv}).$$

□

**Poznámka 4.** *V dôkaze predchádzajúcej vety je ľahko postrehnuteľné, že* $L_{adv}$ *pri akceptácii jazyka* $L^R$ *nijakým spôsobom nepomôže - neuľahčí jeho výpočet. Presnejšie, dodatočná informácia* $L_{adv}$ *o jazyku* $L^R$ *nie je pravdivá* [1]*.*

**Veta 4.** *Existuje* $L_{adv} \in \mathscr{R}$ *taký, že* $\mathscr{L}(L_{adv})$ *nie je uzavretá na komplement.*

*Dôkaz.* Nech dodatočná informácia určujúca triedu $\mathscr{L}(L_{adv})$ je $L_{adv} = \{a^{3m}b^{3n} \mid m, n \in \mathbb{N}\}$. Zoberme si jazyk L z $\mathscr{L}(L_{adv})$, $\mathrm{L} = \{a^{6m}b^{6n} \mid m, n \in \mathbb{N}\}$. Potom platí:

$$w = b^6 a^6 \in L^C \wedge w \notin L_{adv} \Rightarrow L^C \notin \mathscr{L}(L_{adv}).$$

□

**Poznámka 5.** *Doterajšie vlastnosti bolo ľahko rozhodnúť, pretože skúmaná operácia (zreťazenie, resp. iterácia) nám vyrobila nové slovo, ktoré sa nenachádza v* $L(A_{adv})$*. Keďže jazyk, pre ktorý je dodatočná informácia užitočná, vzniká prienikom dodatočnej informácie s nejakým novým jazykom, je zjavné, že dodatočná informácia musí byť nadmnožina jazyka, pre ktorý je užitočná. Inak povedané:*

[1] O informácii $L_{adv}$ povieme, že je nepravdivá vtedy, keď vlastnosť $L_{adv}$ nemá žiadne slovo z jazyka, ktorého akceptovanie chceme zjednodušiť.

*Nech $L_{new}$ je jazyk, ktorý spolu s dodatočnou informáciou $L_{adv}$ definuje $L$, teda*

$$L_{new} \cap L_{adv} = L$$
$$\Rightarrow L \subseteq L_{adv}.$$

*Tento prístup nemôžeme využiť pre skúmanie ďalších operácii, keďže $L_{adv}$ bude stále pravdivá aj pre jazyky vzniknuté príslušnou operáciou (nevzniknú nové slová, môžu len ubudnúť a tie čo zostanú sú podľa predpokladu podmnožinou $L_{adv}$). Pre prípadné dokazovanie neuzavretosti budeme musieť hľadať porušenie druhej podmienky užitočnej dodatočnej informácie, t. j. užitočnosť, či vieme zostrojiť automaty pre $L_{new}$ a $L_{adv}$, také, že oba budú mať menší počet stavov ako $sc(L)$.*

**Veta 5.** *Trieda $\mathscr{L}(L_{adv})$ je uzavretá na $\coprod$ (disjoint union) pre ľubovoľné $L_{adv} \in \mathscr{R}$.*

*Dôkaz.* Nech $L_{adv} \in \mathscr{R}$ a nech $L_1$ a $L_2$ sú ľubovoľné dva jazyky z $\mathscr{L}(L_{adv})$ a $L_3 = L_1 \amalg L_2$. Musíme ukázať, že jazyku $L_3$ informácia $L_{adv}$ dostatočne pomôže. Vieme, že existuje jazyk $L'_1$, resp. $L'_2$ taký, že $L'_1 \cap L_{adv} = L_1$, resp. $L'_2 \cap L_{adv} = L_2$. Nech $A_{L'_1}$, resp. $A_{L'_2}$ je automat akceptujúci $L'_1$, resp. $L'_2$. Z definície dodatočnej informácie vieme, že $A_{L'_1}$, resp. $A_{L'_2}$ má menší počet stavov ako minimálny automat akceptujúci $L_1$, resp. $L_2$. Nech $A_{L_3}$ je minimálny automat akceptujúci $L_3$. Pomocou automatov $A_{L'_1}$ a $A_{L'_2}$ zostrojíme automat $A_{L'_3}$, ktorý bude mať menší počet stavov ako $A_{L_3}$.

$A_{L'_3} = (K, \Sigma, \delta, q_{new}, F)$

- $K = \{q_{new}\} \cup K_{L'_1} \cup K_{L'_2}$
- $\Sigma = \{a, b\} \cup \Sigma_{L'_1} \cup \Sigma_{L'_2}; a, b \notin \Sigma_{L'_1} \cup \Sigma_{L'_2}$
- $F = F_{L'_1} \cup F_{L'_2}$
- $\delta : \delta_{L'_1} \cup \delta_{L'_2} \cup \delta(q_{new}, a) = q_{0_{L'_1}} \cup \delta(q_{new}, b) = q_{0_{L'_2}}$

Stavová zložitosť $A_{L'_3}$ je $\#_s(A_{L'_1}) + \#_s(A_{L'_2}) + 1$. Vidno, že na vytvorenie automatu pre disjoint union treba aspoň jeden nový stav (funkcionalita tohoto stavu je potrebná a nie je možné ju zahrnúť medzi existujúce stavy automatov $A_{L'_1}, A_{L'_2}$. Preto:

$$(\#_s(A_{L'_1}) == sc(L'_1)) \wedge (\#_s(A_{L'_2}) == sc(L'_2)) \Rightarrow \#_s(A_{L'_3}) = sc(L'_3).$$

Platí:

- $\#_s(A_{L'_1}) < \#_s(A_{L_1})$
- $\#_s(A_{L'_2}) < \#_s(A_{L_2})$

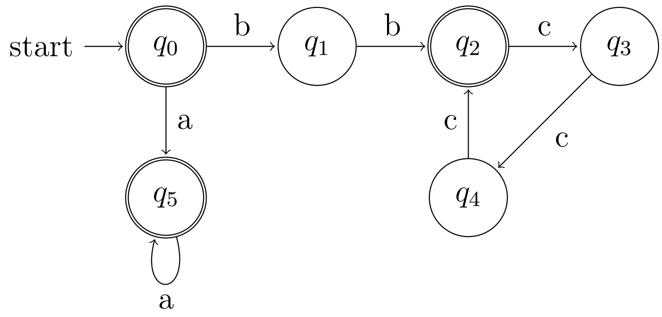
Rovnako aj automat $A_{L_3}$ bude mať počet stavov aspoň o 1 viac oproti $\#_s(A_{L_1}) + \#_s(A_{L_2})$. Preto je automat $A_{L'_3}$ zjednodušený aspoň o dva stavy oproti $A_{L_3}$.

□

**Veta 6.** *Existuje $L_{adv} \in \mathscr{R}$ taký, že $\mathscr{L}(L_{adv})$ nie je uzavretá na prienik.*

*Dôkaz.* Nech $\mathscr{L}(L_{adv})$ je trieda určená dodatočnou informáciou $L_{adv} = \{bb\{c,d\}^{2k} \mid k \geq 1\} \cup a^*$.

Nech $L_1 = \{bbc^{6k} \mid k \geq 1\} \cup a^*$ a $L_2 = \{bbd^{10k} \mid k \geq 1\} \cup a^*$. O užitočnosti dodatočnej informácie $L_{adv}$ pre $L_1$ a $L_2$ je ľahké sa presvedčiť, $L_{1_{new}} = \{bbc^{3k} \mid k \geq 1\} \cup a^*$, podobne $L_{2_{new}}$ pre $L_2$. Ďalej, $L_3 = L_1 \cap L_2$, teda $L_3 = a^*$ Síce dodatočná informácia




Obr. 2.1: Automat pre jazyk $L_{1_{new}}$

je nadmnožina jazyka $L_3$, len ten možno akceptovať jediným stavom. Nemáme preto ako akceptovanie jazyka $L_3$ zjednodušiť, navyše automat pre $L_{adv}$ musí mať viac ako jeden stav. □

**Poznámka 6.** *V dôkaze predchádzajúcej vety, namiesto jazyka $L_{1_{new}}$, môže byť aj jazyk $\{a^*bc^{3k}\} \cup \{a^*\}$, ktorého minimálny automat má o dva stavy menej ako automat pre $L_{1_{new}}$. Prienikom s dodatočnou informáciou si zabezpečíme, že "vymiznú" slová tvaru $a^ibc^{3k}$ pre $i \geq 1$. Prechodový diagram pre automat $A_{L_{1_{new}}}$ je na obrázku 2.1. Podobne pre $L_{2_{new}}$.*

**Veta 7.** *Existuje $L_{adv} \in \mathscr{R}$ taký, že $\mathscr{L}(L_{adv})$ nie je uzavretá na zjednotenie.*

*Dôkaz.* Nech dodatočná informácia je

$$L_{adv} = (\{a\} \cup \{b^j \mid j \equiv 0 \bmod 3\})^* \ .$$

Vezmime si dva konkrétne jazyky z $\mathscr{L}(L_{adv})$,

$$L_1 = (\{a^i \mid i \equiv 0 \bmod 2\} \cup \{b^j \mid j \equiv 0 \bmod 3\})^*$$

a

$$L_2 = (\{a^i \mid i \equiv 1 \bmod 2\} \cup \{b^j \mid j \equiv 0 \bmod 3\})^*.$$

Zjednotením $L_1$ a $L_2$ dostávame jazyk

$$L_3 = (\{a\} \cup \{b^j \mid j \equiv 0 \bmod 3\})^*.$$

Keďže $sc(L_3) = sc(L_{adv})$, nakoľko $L_3 = L_{adv}$, dodatočná informácia nám jazyk $L_3$ nemá ako zjednodušiť. □

**Veta 8.** *Existuje* $L_{adv} \in \mathscr{R}$ *taký, že* $\mathscr{L}(L_{adv})$ *nie je uzavretá na homomorfizmus.*

*Dôkaz.* Nech dodatočná informácia je $L_{adv} = \{a^{2k} \mid k \in \mathbb{N}\}$, homomorfizmus $h(a) = aa$ a nech $L = \{a^{6k} \mid k \in \mathbb{N}\}$. Potom $h(L) = L^{'} = \{a^{12k} \mid k \in \mathbb{N}\}$. Ak by sa dal $L^{'}$ zjednodušiť, musel by existovať jazyk $L_{new}$ taký, že $sc(L_{new})$ je menej ako počet stavov $sc(L^{'}) = 12$ a s prienikom s $L_{adv}$ by musel vytvoriť $L^{'}$. Jazyky na ktoré bola aplikovaná operácia prienik s $L_{adv}$ tvorené DKA s menej ako 12 stavmi nad abecedou $\{a\}$ sú nasledovné:

- $\{a^{11k}\} \cap L_{adv} = \{a^{22k}\}$
- $\{a^{10k}\} \cap L_{adv} = \{a^{10k}\}$
- $\{a^{9k}\} \cap L_{adv} = \{a^{9k}\}$
- $\{a^{8k}\} \cap L_{adv} = \{a^{8k}\}$
- $\{a^{7k}\} \cap L_{adv} = \{a^{14k}\}$
- $\{a^{6k}\} \cap L_{adv} = \{a^{6k}\}$
- $\{a^{5k}\} \cap L_{adv} = \{a^{10k}\}$
- $\{a^{4k}\} \cap L_{adv} = \{a^{4k}\}$
- $\{a^{3k}\} \cap L_{adv} = \{a^{6k}\}$
- $\{a^{2k}\} \cap L_{adv} = \{a^{2k}\}$
- $\{a^{1k}\} \cap L_{adv} = \{a^{*}\}$

Keďže jazyk $L^{'}$ nie je ani jeden z nich, $L^{'} \notin \mathscr{L}(L_{adv})$, je to spor s predpokladaným tvrdením. □

**Veta 9.** *Existuje* $L_{adv} \in \mathscr{R}$ *taký, že* $\mathscr{L}(L_{adv})$ *nie je uzavretá na inverzný homomorfizmus.*

*Dôkaz.* Majme homomorfizmus $h(a) = a^6$, $h(b) = b$ a jazyk

$$L = (\{a^i \mid i \equiv 0 \text{ mod } 6\} \cup \{b^j \mid j \equiv 0 \text{ mod } 3\})^*$$

ktorý je z triedy $\mathscr{L}(L_{adv})$, pričom je táto trieda daná jazykom

$$L_{adv} = (\{a^i \mid i \equiv 0 \text{ mod } 2\} \cup \{b^j \mid j \equiv 0 \text{ mod } 3\})^*.$$

Jazyk vytvorený zobrazením jazyka L inverzným homomorfizmom $h^{-1}$:

$$h^{-1}(L) = L^{'} = (\{a\} \cup \{b^j \mid j \equiv 0 \text{ mod } 3\})^*.$$

Síce je stále možné pomocou dodatočnej informácie $L_{adv}$ zjednodušiť jazyk $L^{'}$, ale platí:

$$sc(L_{adv}) > sc(L^{'})$$

čím už nie je splnený predpoklad pre užitočnosť tejto dodatočnej informácie. □

Preskúmali sme dokopy deväť uzáverových vlastností triedy $\mathscr{L}(L_{adv})$. Na záver pridávame tabuľku 2 s prehľadom výsledkov, ktoré sú, až na množinovú operáciu *disjoint union*, rovnaké.

| **Operácia** | **Symbol** | **Uzavretosť** |
|---|---|---|
| Zreťazenie | $.$ | Nie |
| Iterácia | $^*$ | Nie |
| Reverz | $^R$ | Nie |
| Komplement | $^C$ | Nie |
| Disjoint union | $\coprod$ | Áno |
| Prienik | $\cap$ | Nie |
| Zjednotenie | $\cup$ | Nie |
| Homomorfizmus | $h$ | Nie |
| Inverzný homomorfizmus | $h^{-1}$ | Nie |

Tabuľka 2.1: Uzáverové vlastnosti triedy $\mathscr{L}(L_{adv})$

# Kapitola 3

# Informačná sila jazykov

V tejto kapitole budeme skúmať vzťahy dodatočných informácií v porovnaní so vzťahmi tried určenými týmito informáciami. Pozrieme sa na to, ako sa dvom dodatočným informáciám spoločne pridáva, alebo nepridáva informačná sila. Ukážeme ako rozdielnym jazykom pomáhajú veľmi podobné dodatočné informácie, resp. keď jedna je podmnožinou druhej.

**Veta 10.** *Existujú jazyky $L^a_{adv}, L^b_{adv}$ také, že $L^a_{adv} \subseteq L^b_{adv}$ a $\mathscr{L}(L^a_{adv}) \nsubseteq \mathscr{L}(L^b_{adv})$.*

*Dôkaz.* Majme dva jazyky (dve dodatočné informácie)

- $L^a_{adv} = \{a^{4k} \mid k \in \mathbb{N}\}$
- $L^b_{adv} = \{a^{2k} \mid k \in \mathbb{N}\}$.

Zrejme platí $L^a_{adv} \subseteq L^b_{adv}$. Majme jazyk $L = \{a^{14k} \mid k \in \mathbb{N}\}$. Je ľahko rozhodnúť, že jazyk $L \in \mathscr{L}(L^b_{adv})$. Rovnako zrejmé je aj, že informácia $L^a_{adv}$ nám nevie pomôcť zjednodušiť výpočet jazyka $L$. $\square$

**Poznámka 7.** *Je dobré si uvedomiť, že do triedy jazykov, danou dodatočnou informáciou $L^b_{adv}$ z predchádzajúceho dôkazu, patria len jazyky, ktoré sú podmnožinou $\{a^{2nk} \mid k \in \mathbb{N}\}$, pričom $n \in \mathbb{N}, n \equiv 1 \mod 2$. Teda akonáhle by sme nevedeli a-čka rozdeliť na úseky, ktoré budú dvojnásobok nepárneho čísla, nevieme tento jazyk zjednodušiť (stále pracujeme nad abecedou $\{a\}$, neuvažujeme tu prípad, kedy by sa v slove mohol vyskytovať aj iný symbol rôzny od a).*

**Veta 11.** *Existujú jazyky $L_{adv_1}$, $L_{adv_2}$ také, že $L_{adv_1} \nsubseteq L_{adv_2}$ a $\mathscr{L}(L_{adv_1}) \nsubseteq \mathscr{L}(L_{adv_2})$.*

*Dôkaz.* Majme jazyky $L_{adv_1}$ a $L_{adv_2}$ nasledovné:

- $L_{adv_1} = \{a^{2k} \mid k \in \mathbb{N}\}$
- $L_{adv_2} = \{a^{4k} \mid k \in \mathbb{N}\}$

Zrejme $L_{adv_1} \nsubseteq L_{adv_2}$. Majme jazyk $L = \{a^{6k} \mid k \in \mathbb{N}\}$. Tomuto jazyku vie pomôcť $L_{adv_1}$: $L_{new} = \{a^{3k} \mid k \in \mathbb{N}\}$, $L_{new} \cap L_{adv_1} = L$, teda $L \in \mathscr{L}(L_{adv_1})$. Avšak, $L \nsubseteq L_{adv_2}$, preto $L \notin \mathscr{L}(L_{adv_2})$. □

Na predchádzajúcich vetách, veta 10 a veta 11, je dobre postrehnuteľné správanie dodatočnej informácie. Nezávisle od toho, či je, alebo nie je $L_{adv_1}$ podmnožina $L_{adv_2}$, môžu byť $L_{adv_1}$ a $L_{adv_2}$ dodatočnými informáciami pre rozdielne jazyky.

V nasledujúcej časti sa pozrieme na nárast, alebo pokles informačnej sily, keď spravíme prienik, alebo zjednotenie dodatočných informácií. Pojem 'informačná sila' nebudeme nijak formalizovať a ponecháme ho na intuitívnom porozumení čitateľa. V podstate nárast/pokles informačnej sily hovorí to, že o koľko viac/menej jazykom vieme pomôcť zjednodušiť akceptačný výpočet, keď informačná sila narastie/poklesne.

**Veta 12.** *Existujú jazyky $L_1$ a $L_2$ z $\mathscr{R}$ také, že*

$$\mathscr{L}(L_1) \cap \mathscr{L}(L_2) \nsubseteq \mathscr{L}(L_1 \cap L_2).$$

*Dôkaz.* Úvaha je podobná ako v dôkaze neuzavretosti triedy $\mathscr{L}(L_{adv})$ na prienik z predchádzajúcej kapitoly. Pokiaľ prienikom dodatočných informácií $L_1$ a $L_2$ vznikne jazyk, ktorý možno akceptovať automatom s jedným stavom, tento jazyk nie je užitočnou informáciou pre akýkoľvek iný jazyk. Potom by do $\mathscr{L}(L_1 \cap L_2)$ nepatril žiadny jazyk a do $\mathscr{L}(L_1) \cap \mathscr{L}(L_2)$ budú patriť jazyky ktorým $L_1$ a $L_2$ pomôže aspoň každý samostatne. Teda chceme nájsť $L_1$ a $L_2$ také, že budú dodatočnými informáciami pre nejaký jazyk (nie nutne rovnaký) a ich prienikom vznikne jazyk, ktorý už inému jazyku nepomôže. Také jazyky $L_1$ a $L_2$ existujú a sú to napr. jazyky použité v dôkaze vety 6. □

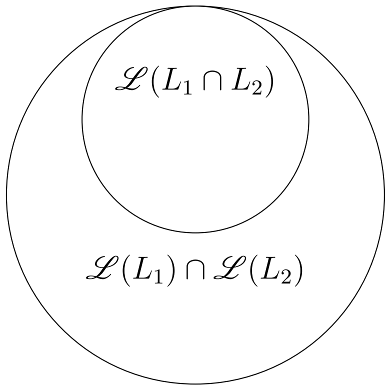
**Veta 13.** *Nech $L_1$ a $L_2$ sú jazyky z $\mathscr{R}$. Potom*

$$\mathscr{L}(L_1 \cap L_2) \subseteq \mathscr{L}(L_1) \cap \mathscr{L}(L_2).$$

*Dôkaz.* Pokiaľ $\mathscr{L}(L_1 \cap L_2) = \emptyset$, potom platí $\mathscr{L}(L_1 \cap L_2) \subseteq \mathscr{L}(L_1) \cap \mathscr{L}(L_2)$. V opačnom prípade, nech $L$ je ľubovoľný jazyk z $\mathscr{L}(L_1 \cap L_2)$. Z definície vieme, že $L \subsetneq L_1 \cap L_2$. Preto platí $L \subseteq L_1$ a zároveň $L \subseteq L_2$. Potrebujeme ukázať, že existujú jazyky $L_1^{'} \in \mathscr{L}(L_1)$ a $L_2^{'} \in \mathscr{L}(L_2)$, také, že

$$L_1^{'} = L \cup L_{1_{zvyok}}$$
$$L_2^{'} = L \cup L_{2_{zvyok}}$$
$$L_{1_{zvyok}} \cap L_{2_{zvyok}} = \emptyset$$

a teda: $L = L_1^{'} \cap L_2^{'}$. Jazyk $L_1^{'}$ patrí do $\mathscr{L}(L_1)$, pretože obsahuje všetky slová, ktoré sú v $L$ a sú to práve tie, akým vedel (aj) $L_1$ pomôcť zjednodušiť akceptačný výpočet (prípadne ešte nejakým iným, ktoré sú v $L_{1_{zvyok}}$). Pre $L_2$ podobne. A preto $L \in \mathscr{L}(L_1) \cap \mathscr{L}(L_2)$. Vzťah triedy $\mathscr{L}(L_1 \cap L_2)$ k prieniku tried $\mathscr{L}(L_1)$ a $\mathscr{L}(L_2)$ je znázornený na obrázku 3.1. □




Obr. 3.1: Vzťah $\mathscr{L}(L_1) \cap \mathscr{L}(L_2)$ s triedou $\mathscr{L}(L_1 \cap L_2)$

**Veta 14.** *Existujú jazyky* $L_1$ *a* $L_2$ *z* $\mathscr{R}$ *také, že*

$$\mathscr{L}(L_1) \cup \mathscr{L}(L_2) \not\subseteq \mathscr{L}(L_1 \cup L_2).$$

*Dôkaz.* Majme jazyky

$$L_1 = (\{a^i \mid i \equiv 0 \bmod 2\} \cup \{b^j \mid j \equiv 0 \bmod 3\})^*$$

a

$$L_2 = (\{a^i \mid i \equiv 1 \bmod 2\} \cup \{b^j \mid j \equiv 0 \bmod 3\})^*$$

určujúce triedy $\mathscr{L}(L_1)$, $\mathscr{L}(L_2)$ a $\mathscr{L}(L_1 \cup L_2)$. Jazyk $L_1$ si môžeme predstaviť ako slová, v ktorom sa striedajú úseky znakov $a$ a $b$, pričom dĺžka každého úseku $a$-čok je párna a dĺžka každého úseku $b$-čok je deliteľná tromi. Jazyk $L_2$ podobne, s tým, že dĺžka každého úseku $a$-čok je nepárna. Nech jazyk $L_{1,2}$ vznikne prienikom jazykov $L_1$ a $L_2$,

$$L_{1,2} = (a^* \cup \{b^j \mid j \equiv 0 \bmod 3\})^*.$$

Tento jazyk (dodatočná informácia) je jednoduchší a nepomôže tam, kde bude treba kontrolovať párnosť znakov $a$ v každom úseku (jazyku ešte stále môže pomôcť, ale len pre úseky znakov $b$). Preto jazyk $L = \{a^i \mid i \equiv 0 \bmod 6\}$ nepatrí do $\mathscr{L}(L_1 \cup L_2)$. Síce platí $sc(L_{1,2}) < sc(L)$, ale $L_{1,2}$ nezjednoduší výpočet $L$, presnejšie

$$\nexists L_{new} : L_{new} \cap L_{1,2} = L \wedge sc(L_{new}) < sc(L)$$

Avšak $L$ patrí do $\mathscr{L}(L_1)$ a teda aj do $\mathscr{L}(L_1) \cup \mathscr{L}(L_2)$. □

**Veta 15.** *Existujú jazyky* $L_1$ *a* $L_2$ *z* $\mathscr{R}$ *také, že*

$$\mathscr{L}(L_1 \cup L_2) \nsubseteq \mathscr{L}(L_1) \cup \mathscr{L}(L_2).$$

*Dôkaz.* Nech $L_1 = \{a^{3k} \mid k \in \mathbb{N}\}$ a nech $L_2 = \{b^{3k} \mid k \in \mathbb{N}\}$. Ďalej, nech zjednotením $L_1$ a $L_2$ vznikne jazyk $L$:

$$L = L_1 \cup L_2 = \{a^{3k} \mid k \in \mathbb{N}\} \cup \{b^{3k} \mid k \in \mathbb{N}\}$$

Majme jazyk $L^{'} = \{a^{6k} \mid k \in \mathbb{N}\} \cup \{b^{6k} \mid k \in \mathbb{N}\}$. Platí: $L^{'} \in \mathscr{L}(L)$. Môžeme si to predstaviť tak, že nezávisle od toho či bude mať automat pre $L^{'}$ na vstupe $a$-čka alebo $b$-čka (správnej násobnosti), vie mu automat pre $L$ zjednodušiť výpočet, keďže obsahuje všetky tieto slová. Avšak, $L^{'} \notin \mathscr{L}(L_1) \cup \mathscr{L}(L_2)$. Platí $L^{'} \notin \mathscr{L}(L_1)$, lebo $\nexists L_{new} : L^{'} = L_{new} \cap L_1$, kvôli tomu, že $L^{'} \nsubseteq L_1$. Z rovnakého dôvodu $L^{'} \notin \mathscr{L}(L_2)$. A teda $L^{'} \notin \mathscr{L}(L_1) \cup \mathscr{L}(L_2)$. □

V predchádzajúcich štyroch vetách sme dokázali vzťah množinového usporiadania $\mathscr{L}(L_1 \cap L_2)$ s $\mathscr{L}(L_1) \cap \mathscr{L}(L_2)$ a vzťah $\mathscr{L}(L_1 \cup L_2)$ s $\mathscr{L}(L_1) \cup \mathscr{L}(L_2)$. Pomerne prirodzený vzťah je, že $\mathscr{L}(L_1 \cap L_2) \subseteq \mathscr{L}(L_1) \cap \mathscr{L}(L_2)$, keďže prienikom sa môžu slová oproti $L_1$ a $L_2$ len "stratiť" a teda dodatočná informácia $L_1 \cap L_2$ nepomôže viac jazykom (nemôže narásť jej informačná sila) ako tým, ktoré patria do triedy $\mathscr{L}(L_1) \cap \mathscr{L}(L_2)$. Pre prehľadnosť, tieto výsledky uvádzame v tabuľke 3.

| | | |
|---|---|---|
| $\mathscr{L}(L_1 \cap L_2)$ | $\not\supseteq$ <br> $\subseteq$ | $\mathscr{L}(L_1) \cap \mathscr{L}(L_2)$ |
| $\mathscr{L}(L_1 \cup L_2)$ | $\not\supseteq$ <br> $\nsubseteq$ | $\mathscr{L}(L_1) \cup \mathscr{L}(L_2)$ |

Tabuľka 3.1: Porovnanie informačnej sily

# Záver

V tejto práci sme skúmali vlastnosti tried jazykov určených konkrétnou dodatočnou informáciou, ktorá je navyše užitočná.

Dodatočná informácia daná deterministickým konečným automatom, pomáha zjednodušiť akceptačné výpočty slov pre nejaké jazyky. Všetky jazyky, ktorým vie konkrétna dodatočná informácia pomôcť, tvoria triedu jazykov.

Nadviazali sme tým na práce [1], [3] a [4], ktoré sa zaoberali skúmaním užitočnosti informácie na rôznych výpočtových modeloch.

Pozreli sme sa na uzáverové vlastnosti spomenutej triedy jazykov. Z deväť skúmaných množinových operácií, bola táto trieda uzavretá iba na *disjoint union*.

Skúmali sme aj informačnú silu jazykov tak, že sme porovnávali vzťahy medzi informáciami so vzťahmi medzi nimi určenými triedami jazykov. Ukázali sme, ako rozdielnej triede jazykov pomáha nejaká dodatočná informácia a od nej konkrétnejšia dodatočná informácia, ktorá je jej podmnožinou. Nakoniec sme sa pozreli na to, ako narastie, resp. ako sa stratí informačná sila dodatočných informácií, keď spravíme ich prienik alebo zjednotenie.

Jedným z možných pokračovaní tejto práce môže byť zadefinovanie triedy jazykov určenou dodatočnou informáciou na inom výpočtovom modeli ako sú deterministické konečné automaty.

# Literatúra

[1] P. Gaži and B. Rovan. Assisted problem solving and decompositions of finite automata. *SOFSEM 2008. LNCS, vol. 4910, Springer, Heidelberg*, 2008.

[2] J. E. Hopcroft, R. Motwani, and J. D. Ullman. *Introduction to Automata Theory, Languages and Computation (3rd Edition)*. Addison-Wesley Longman Publishing Co., Inc., 2006.

[3] P. Labath and B. Rovan. Simplifying DPDA using supplementary information. *LATA 2011. LNCS, vol 6638. Springer, Berlin, Heidelberg*, 2011.

[4] Š. Sádovský. Prídavná informácia a zložitosť nedeterministických konečných automatov. Diplomová práca, Univerzita Komenského v Bratislave, 2017.