Univerzita Komenského v Bratislave
Fakulta Matematiky, Fyziky a Informatiky
Katedra Informatiky

O využívaní stavov v konečných automatoch

(bakalárska práca)

Ivan Kováč

Vedúci: prof. RNDr. Branislav Rovan, PhD.

Bratislava, 2008

Katedra Informatiky
Fakulta Matematiky, Fyziky a Informatiky
Univerzita Komenského v Bratislave

O využívaní stavov v konečných automatoch

(bakalárska práca)

Ivan Kováč

Študijný odbor: Informatika 9.2.1

Vedúci: prof. RNDr. Branislav Rovan, PhD.

Bratislava, 2008

Čestne prehlasujem, že som túto bakalársku prácu vypracoval samostatne s použitím citovaných zdrojov.

. . . .

Poďakovanie

V prvom rade by som chcel poďakovať vedúcemu bakalárskej práce profesorovi Branislavovi Rovanovi za zaujímavú tému a cenné rady pri vypracovávaní práce.

Tiež by som chcel poďakovať svojej priateľke za trpezlivosť a rodine za všetko.

Abstrakt

Autor:	Ivan Kováč
Názov bakalárskej práce:	O využívaní stavov v konečných automatoch
Škola:	Univerzita Komenského v Bratislave
Fakulta:	Fakulta matematiky, fyziky a informatiky
Katedra:	Katedra informatiky
Vedúci bakalárskej práce:	prof. RNDr. Branislav Rovan, PhD.
Rozsah práce:	30 strán

Bratislava, jún 2008

V práci sa zaoberáme rovnomerným využívaním stavov deterministických konečných automatov. Prinášame analýzu niekoľkých možných definícií pojmu „rovnomerné využívanie stavov“. Ukážeme vzťah medzi minimálnym automatom v zmysle Myhill-Nerodovej vety a rovnomerne využívanými automatmi. Ďalej pracujeme s triedou jazykov, pre ktoré existuje rovnomerne využívaný automat, uvádzame základné uzáverové vlastnosti tejto triedy a zaraďujeme ju do Chomského hierarchie.

Kľúčové slová: deterministické konečné automaty, využívanie stavov, minimálny automat.

Predhovor

V dnešnej modernej dobe sa čoraz častejšie stretávame s počítačmi aj na miestach, kde by sme to v minulosti neočakávali. Preto sa snažíme vylepšovať jednotlivé časti počítača, zefektívniť ich prácu, ale aj znížiť ich nepriaznivý vplyv na životné prostredie. Súčasťou každého počítača je procesor – riadiaca jednotka, ktorá vykonáva výpočet.

Zaujímavou otázkou spojenou s vývojom procesorov je ich zahrievanie počas výpočtu. (Hlavne výpočtovo náročné úlohy zamestnávajú procesor natoľko, že ho treba chladiť.) Tento problém sa síce v praxi rieši, no pre potreby teórie doposiaľ nebol vypracovaný žiaden zodpovedajúci model. Preto v práci prinášame analýzu využívania stavov konečných automatov. (Konečné automaty považujeme za jeden z najjednoduchších modelov procesora.) Keďže je našou snahou dosiahnuť rovnomerné zahrievanie procesorov, budeme uvažovať *rovnomerné* využívanie stavov automatov.

Prvým cieľom práce bola analýza rôznych definícií rovnomerného využívania automatov. Mnoho ľudí si iste pod rovnomerným využívaním stavov automatu predstaví niečo iné, preto sme týchto definícií navrhli viacero, porovnali ich z pohľadu jednoduchosti a „sily“, no tiež sme dbali na to, aby definície príliš neodbočovali od našej intuitívnej predstavy rovnomerne využívaných automatov.

Po vybratí vhodnej definície bolo našim hlavným cieľom bližšie určiť triedu jazykov, ku ktorým existujú rovnomerne využívané automaty. K tomu sme si v práci vybrali skúmanie uzáverových vlastností tejto triedy, čím sme triedu jasne odlíšili od triedy regulárnych jazykov.

Obsah

Kapitola 1

Úvod

Pojem deterministického konečného automatu je v teórii formálnych jazykov a automatov dobre známy. Konečný automat – jeden z najjednoduchších typov konečnostavových zariadení – môže byť vnímaný ako zjednodušený teoretický model pre procesor počítača. Medzi problémy spojené s vývojom procesora určite patrí aj jeho zahrievanie. Počas výpočtu procesor mení obsah svojich registrov a pri každej takejto zmene vyprodukuje istú dávku tepla, kvôli ktorému potom treba procesor chladiť. (Ak by sa kremík na čipe roztavil, pravdepodobne by procesor už nefungoval tak, ako by mal fungovať.) Môže sa stať, že sa procesor príliš zahrieva na malej časti a na iných miestach je v podstate chladný. Za predpokladu, že sa procesor bude zahrievať rovnomerne, by bolo možné chladiť procesor efektívnejšie. Otázkou zostáva, či sa procesor dá navrhnúť tak, aby sa zahrieval rovnomerne a zároveň aby nestratil svoju výpočtovú silu.

V práci budeme skúmať zjednodušený matematický model procesoru, deterministický konečný automat. Budeme predpokladať, že krok výpočtu automatu spôsobí zahriatie istej časti procesoru – v závislosti od aktuálneho stavu automatu[1], a teda budeme konštruovať automaty, ktoré rovnomerne využívajú svoje stavy. Téme rovnomerného využívania stavov automatov – teoretického riešenia problému so zahrievaním sa procesorov – sa doposiaľ nevenovala pozornosť. Preto v práci začneme návrhom definície rovnomerného využívania stavov.

V kapitole 2 pre úplnosť uvedieme základné pojmy, s ktorými budeme v ďalšom texte pracovať – definíciu deterministického konečného automatu, niektoré operácie na jazykoch. Predpokladáme však, že čitateľ absolvoval aspoň základný kurz teórie formálnych jazykov

[1]Tento predpoklad nie je úplne správny – v praxi je zahrievanie procesora zložitejší proces. Vzhľadom na to, že budeme skúmať zjednodušený model, veríme, že nám čitateľ túto nepresnosť odpustí.

a automatov a má predstavu o týchto pojmoch, preto budeme struční. V kapitole 3 zadefinujeme pojem rovnomerne využívaného automatu viacerými spôsobmi a porovnáme tieto definície. Vyberieme definíciu, ktorá podľa nás najlepšie vystihuje intuitívne chápanie rovnomerného využívania stavov. S touto definíciou budeme pracovať v ďalšom texte. V Kapitole 4 potom budeme skúmať triedu jazykov akceptovaných konečnými automatmi, ktoré rovnomerne využívajú svoje stavy. Dokážeme vzťah medzi minimálnym automatom a rovnomerne využívanými automatmi pre daný jazyk. Uvedieme uzáverové vlastnosti tejto triedy a naznačíme, ktoré vlastnosti by sa zmenili inou voľbou definície.

Vhodná voľba definície nie je jednoduchá úloha. Skúsime nájsť kompromis medzi dostatočnou presnosťou pojmu (tak, aby definícia vystihovala naše intuitívne vnímanie rovnomerného využívania stavov) a dobrými vlastnosťami definície. Musíme však pamätať aj na to, že teoretický model nemôže presne vystihnúť situáciu pozorovanú v praxi.

Kapitola 2

Základné pojmy

V celej práci sa zaoberáme deterministickými konečnými automatmi. Uvádzame preto definíciu tohto pojmu ako je uvedená v [HMU01]. Pre úplnosť uvedieme aj definíciu nedeterministického konečného automatu, aj keď sa ním táto práca nezaoberá. (Výsledky môžu byť pri vhodnej voľbe definície rovnomerného využívania stavov rozšíriteľné aj na nedeterministické automaty.) Čitateľovi, ktorý nie je oboznámený s pojmami *slovo*, *jazyk*, *trieda jazykov*, ako aj s ostatnými základnými pojmami teórie formálnych jazykov, odporúčame naštudovať príslušné pojmy napríklad v [RF08].

2.1 Deterministický konečný automat

Definícia 2.1.1. *Deterministický konečný automat (DKA) je pätica $A = (K, \Sigma, \delta, q_0, F)$, kde K je konečná množina stavov, Σ je konečná vstupná abeceda, $q_0 \in K$ je počiatočný stav, $F \subseteq K$ je množina akceptačných stavov a $\delta : K \times \Sigma \to K$ je prechodová funkcia.*

Definícia 2.1.2. *Konfigurácia DKA A je dvojica $(q, w) \in K \times \Sigma^*$, kde q je stav automatu a w je nespracovaná časť vstupného slova.*

Definícia 2.1.3. *Krok výpočtu DKA A je relácia $\vdash_A$ na konfiguráciach definovaná nasledovne:*

$$(q, av) \vdash_A (p, v) \iff p = \delta(q, a).$$

Definícia 2.1.4. *Jazyk akceptovaný deterministickým konečným automatom A je množina slov $L(A) = \{w \in \Sigma^* \mid (q_0, w) \vdash_A^* (q, \varepsilon) \wedge q \in F\}$.*

Poznámka 2.1.1. *V predchádzajúcej definícii sme použili dve štandardné označenia. Jazyk* Σ^* *označuje jazyk (množina) všetkých slov nad abecedou* Σ*. Relácia* $\vdash_A^*$ *označuje reflexívno-tranzitívny uzáver relácie* $\vdash_A$*.*

2.1.1 Deterministický konečný automat so zastavením

Pre potreby práce budeme uvažovať miernu obmenu deterministického konečného automatu. Jediný rozdiel medzi klasickou definíciou a definíciou používanou v práci bude to, že klasický deterministický automat musí vždy dočítať slovo, nemôže sa zastaviť a neakceptovať slovo. Úpravou prechodovej funkcie zabezpečíme, aby sa automat mohol zaseknúť. (A tým neakceptovať slovo, na ktorom sa zasekne.) Nejjednoduchšou úpravou δ-funkcie je povoliť, aby bola na niektorých dvojiciach stav-písmeno nedefinovaná. Napriek tomu, že takto definovaná prechodová funkcia už nie je funkciou (nie je definovaná na všetkých prvkoch), ale reláciou, budeme stále hovoriť o prechodovej funkcii.

Poznámka 2.1.2. *Krok výpočtu, konfigurácia a akceptačný jazyk sú definované analogicky ako pri klasickom DKA.*

Čitateľ si môže všimnúť, že takto definovaný deterministický konečný automat má rovnakú výpočtovú silu ako kalsický DKA – Zaseknutie sa automatu je ekvivalentné s prechodom do stavu q_u, ktorý je neakceptačný, a v ktorom je pre každé písmeno x z abecedy prechodová funkcia definovaná ako $\delta(q_u, x) = q_u$.

Napriek tomu, že budeme používať DKA, ktorý dokáže počas výpočtu zastaviť, budeme ho nazývať (deterministický konečný) automat. V niektorých prípadoch túto skutočnosť pripomenieme v poznámkach pod čiarou.

2.2 Nedeterministický konečný automat

Definícia 2.2.1. *Nedeterministický konečný automat (NKA) je pätica* $A = (K, \Sigma, \delta, q_0, F)$*, kde* K *je konečná množina stavov,* Σ *je konečná vstupná abeceda,* $q_0 \in K$ *je počiatočný stav,* $F \subseteq K$ *je množina akceptačných stavov a* $\delta : K \times (\Sigma \cup \{\varepsilon\}) \to 2^K$ *je prechodová funkcia.*

Definícia 2.2.2. *Konfigurácia NKA* A *je dvojica* $(q, w) \in K \times \Sigma^*$*, kde* q *je stav automatu a* w *je nespracovaná časť vstupného slova.*

Definícia 2.2.3. *Krok výpočtu NKA A je relácia $\vdash_A$ na konfiguráciach definovaná nasledovne:*

$$(q, av) \vdash_A (p, v) \iff p \in \delta(q, a).$$

Definícia 2.2.4. *Jazyk akceptovaný nedeterministickým konečným automatom A je množina slov $L(A) = \{w \in \Sigma^* \mid (q_0, w) \vdash_A^* (q, \varepsilon) \wedge q \in F\}$.*

2.3 Operácie s jazykmi

V práci budeme hovoriť aj o uzáverových vlastnostiach triedy jazykov prislúchajúcich k rovnomerne využívaným automatom vzhľadom na operácie s jazykmi. Predpokladáme, že operácie *prienik*, *zjednotenie* a *komplement*[1] sú čitateľovi známe aj z teórie množín. Uvedieme prehľad operácií, ktoré sa používajú v teórií formálnych jazykov.

Označené zjednotenie $L_1 \cup_c L_2 = L_1 \cup \{c\} \cdot L_2$, pričom $c \notin \Sigma_{L_1}$ $c \notin \Sigma_{L_2}$.

Zreťazenie $L_1 \cdot L_2 = \{u \cdot v \mid u \in L_1, v \in L_2\}$, kde ak $u = u_1u_2 \dots u_i$, $v = v_1v_2 \dots v_j$, tak $u \cdot v = u_1u_2 \dots u_iv_1v_2 \dots v_j$.

Mocnina $L^0 = \{\varepsilon\}$ a $\forall n \geq 1; L^n = L \cdot L^{n-1}$.[2]

Iterácia (Kleeneho uzáver $*$) $L^* = \bigcup_{i=0}^{\infty} L^i$.

Kladná iterácia (Kleeneho uzáver $+$) $L^+ = \bigcup_{i=1}^{\infty} L^i$.

Reverz Nech $u = u_1u_2 \dots u_i$. $L^R = \{u^r \mid u \in L\}$, kde $u^R = u_iu_{i-1} \dots u_2u_1$

Homomorfizmus Zobrazenie $h : \Sigma_1^* \to \Sigma_2^*$ také, že $\forall u, v \in \Sigma_1^* : h(u \cdot v) = h(u) \cdot h(v)$, nazývame homomorfizmus. Nech $L \subseteq \Sigma_1^*$, potom $h(L) = \{h(u) \mid u \in L\}$.

Inverzný homomorfizmus Nech h je homomorfizmus. Pomocou h definujeme zobrazenie $h^{-1} : \Sigma_2^* \to 2^{\Sigma_1^*}$ ako $h^{-1}(u) = \{v \mid h(v) = u\}$.[3] Potom $h^{-1}(L) = \bigcup_{w \in L} h^{-1}(w)$.

Substitúcia Zobrazenie $\varphi : \Sigma_1^* \to 2^{\Sigma_2^*}$ také, že $\forall u, v \in \Sigma_1^* : \varphi(u \cdot v) = \varphi(u) \cdot \varphi(v)$, nazývame substitúcia. Nech $L \subseteq \Sigma_1^*$, potom $\varphi(L) = \bigcup_{w \in L} \varphi(w)$.

[1] Pri komplemente jazyka L považujeme za univerzálnu množinu jazyk Σ_L^*, kde Σ_L je minimálna abeceda, ktorá obsahuje všetky písmená z jazyka L

[2] Mocnina sa dá analogicky definovať aj pre slová: $w^0 = \varepsilon$ a $\forall n \geq 1; w^n = w \cdot w^{n-1}$

[3] Toto zobrazenie je v istom zmysle inverzné k homomorfizmu h.

Poznámka 2.3.1. *V kapitole 4 budeme skúmať uzáverové vlastnosti triedy rovnomerných jazykov – trieda je uzavretá vzhľadom na operáciu, ak po prevedení operácie s jazykom z triedy vznikne tiež jazyk z tejto triedy. Pri uzavretosti triedy jazykov $\mathcal{T}$ vzhľadom na nejakú operáciu pracujúcu s dvomi a viacerými jazykmi budeme predpokladať, že všetky jazyky patria do triedy $\mathcal{T}$.*

Kapitola 3

Definícia rovnomerného využívania stavov DKA

V tejto kapitole si budeme všímať využívanosť stavov DKA A na jazyku $L(A)$ alebo jeho podmnožinách. Preto ak budeme hovoriť o využívaní stavov na jazyku L, budeme predpokladať, že $L \subseteq L(A)$. Ako sme uviedli v kapitole 2, budeme uvažovať deterministického konečného automatu ktorý sa môže počas výpočtu zaseknúť. (Keďže v práci skúmame iba výpočty na slovách z akceptačného jazyka, nebol by stav, ktorý by sa používal len na dočítanie slova a neakceptovanie, nikdy využitý. Potom by automat nemohol rovnomerne využívať svoje stavy – jeden zo stavov by sa nevyužíval vôbec.)

V nasledujúcom texte skúsime vysloviť prirodzenú definíciu rovnomerného využívania stavov DKA a objasniť dôvody, prečo takáto definícia nespĺňa naše požiadavky. Neskôr sa dopracujeme k definíciam, ktoré budú vhodnejšie na využitie. Našim cieľom bude rozpoznať rozdiely medzi definíciami a vybrať tú, ktorá najlepšie vystihne pojem rovnomerného využívania stavov.

Budeme hovoriť o rovnomernom využívaní automatu A na jazyku L. Ak jazyk L nešpecifikujeme, budeme predpokladať, že $L = L(A)$.

3.1 Intuitívna definícia

Na to aby sme o stavoch DKA $A = (K, \Sigma, \delta, q_0, F)$ mohli povedať, že sú rovnomerne využívané na jazyku L, potrebujeme definovať rovnomerné využívanie stavov na slovách. Uvažujme slovo $w = w_1 w_2 \dots w_n \in L$. Pre toto slovo existuje jednoznačný výpočet automatu A

$$(q_0, w_1w_2\dots w_n) \vdash (q_1, w_2w_3\dots w_n) \vdash^* (q_F, \varepsilon).$$

Pre zjednodušenie zápisu zavedieme označenie $\#_A[q,w]$, ktoré bude znamenať počet výskytov stavu q vo výpočte automatu A na slove w. V prípade, že bude jednoznačne určený automat, o ktorom hovoríme, budeme využívať aj alternatívne označenie $\#[q,w]$.

Definujme rovnomerné využívanie stavov na slove w. Prirodzeným spôsobom dostávame nasledujúcu definíciu.

Definícia 3.1.1. *Deterministický konečný automat* $A = (K, \Sigma, \delta, q_0, F)$ *nazveme* rovnomerne využívaný *na slove* $w \in L(A)$, *ak*

$$\forall p, q \in K \qquad \#[p,w] = \#[q,w].$$

Základným problémom tejto definície je fakt, že automat môže byť rovnomerne využívaný na slove w iba ak platí $|w| + 1 = k \cdot |K|$ pre nejaké $k \in \mathbb{N}$. Aby sme sa problémom tohto typu vyhli, môžeme do definície 3.1.1 vniesť parameter, ktorý nám umožní istú nerovnomernosť využívania stavov. Aby sme mohli stále hovoriť o rovnomernom využívaní, povolíme iba konštantnú odchýlku.

Definícia 3.1.2. *Deterministický konečný automat* $A = (K, \Sigma, \delta, q_0, F)$ *nazveme* rovnomerne využívaný *na slove* $w \in L(A)$ *s parametrom* $k \in \mathbb{N}$ *ak*

$$\forall p, q \in K \qquad |\#[p,w] - \#[q,w]| \le k.$$

Týmto parametrom sme sa zbavili obmedzenia na dĺžku slov, na ktorých môžu byť automaty rovnomerne využívané. Vyslovme teraz definíciu rovnomerného využívania automatu na jazykoch. Využijeme pritom definíciu 3.1.2.

Definícia 3.1.3. *Deterministický konečný automat* $A = (K, \Sigma, \delta, q_0, F)$ *nazveme* rovnomerne využívaný *na jazyku* $L \subseteq L(A)$ *s parametrom* $k \in \mathbb{N}$ *ak*

$$\forall w \in L \quad \forall p, q \in K \qquad |\#[p,w] - \#[q,w]| \le k.$$

Príklad 3.1.1. *Pre jazyk* $L_1 = \{a^n \mid n \equiv 3 \mod 7\}$ *existuje rovnomerne využívaný automat* A_1 *taký, že* $L(A_1) = L_1$. *Pritom parameter z definície je rovný* 1. *Aj pre jazyk* $L_2 = \{a^n \mid n \equiv 8 \mod 9\}$ *existuje rovnomerne využívaný automat* A_2 *taký, že* $L(A_2) = L_2$.

Pritom parameter $k = 0$, čo znamená, že automat A_2 je rovnomerne využívaný aj podľa definície 3.1.1 na každom slove z L_2. Prvý automat je

$$A_1 = (\{q_0, q_1, q_2, q_3, q_4, q_5, q_6\}, \{a\}, \delta_1, q_0, \{q_3\}),$$

kde $\delta_1(q_i, a) = q_{i+1 \mod 7}$ pre $i \in \{0, 1, 2, 3, 4, 5, 6\}$. Druhý automat je

$$A_2 = (\{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8\}, \{a\}, \delta_2, q_0, \{q_8\}),$$

kde $\delta_2(q_i, a) = q_{i+1 \mod 9}$ pre $i \in \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$.

Ľahko nahliadneme ako vyzerá výpočet oboch automatov, a tak aj zistíme príslušné konštanty z definície 3.1.3.

V definíciách 3.1.2 a 3.1.3 sme zaviedli istý parameter – odchýlku od skutočne rovnomerného využívania stavov. Parametrov ovplyvňujúcich vlastnosti definícií objavíme v ďalšom texte viac. Budeme skúmať ich vplyv na definíciu a skúsime vytvoriť definíciu, ktorá bude mať podľa nás vyhovujúce vlastnosti.

3.2 Porovnávanie stavov

Vo všetkých troch definíciach sme porovnávali každú dvojicu stavov. V praxi by mohlo byť nepohodlné porovnávanie využívania všetkých stavov navzájom.Naše úvahy zjednoduší, ak budeme porovnávať maximálne a minimálne využívaný stav. Ukážeme, že nasledujúce definície sú ekvivalentné k definíciam 3.1.2 a 3.1.3.

Definícia 3.2.1. *Deterministický konečný automat $A = (K, \Sigma, \delta, q_0, F)$ nazveme* rovnomerne využívaný *na slove $w \in L(A)$ s parametrom $k \in \mathbb{N}$ ak*

$$\max_{p \in K}(\#[p, w]) - \min_{q \in K}(\#[q, w]) \leq k.$$

Definícia 3.2.2. *Deterministický konečný automat $A = (K, \Sigma, \delta, q_0, F)$ nazveme* rovnomerne využívaný *na jazyku $L \subseteq L(A)$ s parametrom $k \in \mathbb{N}$ ak*

$$\forall w \in L \quad \max_{p \in K}(\#[p, w]) - \min_{q \in K}(\#[q, w]) \leq k.$$

Výhodou definície 3.2.2 oproti 3.1.3 je fakt, že v definícii 3.2.2 porovnávame využívanosť len dvoch stavov.

Veta 3.2.1. *Deterministický konečný automat A je rovnomerne využívaný podľa definície 3.1.3 parametrom $k \in \mathbb{N}$ práve vtedy, keď je rovnomerne využívaný podľa definície 3.2.2*

Dôkaz. Ak pre každú dvojicu stavov DKA A platí $|\#[p,w] - \#[q,w]| \leq k$ tak táto nerovnosť platí aj vtedy, ak stav p je maximálne využívaný a stav q minimálne využívaný, teda $\max_{p \in K}(\#[p,w]) - \min_{q \in K}(\#[q,w]) \leq k$. Z toho vyplýva, že DKA A je rovnomerne využívaný aj podľa definície 3.2.2.

Opačne, ak platí $\max_{p \in K}(\#[p,w]) - \min_{q \in K}(\#[q,w]) \leq k$, tak zrejme platí nerovnosť $|\#[p,w] - \#[q,w]| \leq k$ pre každú dvojicu stavov p a q. Ak by totiž pre nejaké dva stavy platilo $\#[p,w] - \#[q,w] > k$, tak využitím nerovností $\#[p,w] \leq \max_{p \in K}(\#[p,w])$ a $\#[q,w] \geq \min_{q \in K}(\#[q,w])$ dostávame $\max_{p \in K}(\#[p,w]) - \min_{q \in K}(\#[q,w]) > k$. Teda ak je DKA A rovnomerne využívaný podľa definície 3.2.1, tak je rovnomerne využívaný aj podľa definície 3.1.1. □

3.3 Parametre definície

V definícii 3.2.2 sme použili parameter k, ktorý nám umožní využívať niektoré stavy viac ako iné pri zachovaní rovnomernosti využívania stavov. Definíciu však ovplyvňuje viacero faktorov. Uvedieme prehľad tých základných.

3.3.1 Voľba stavov

Za automat, ktorý rovnomerne využíva svoje stavy, môžeme považovať aj taký, ktorý využíva rovnomerne všetky stavy okrem počiatočného a ten využíva relatlívne málo. Príkladom takého automatu je automat pre označené zjednotenie dvoch jazykov, pričom tieto jazyky majú rovnomerne využívaný automat podľa definície 3.2.2. Takýto automat však podľa definície 3.2.2 nevyužíva svoje stavy rovnomerne. Túto definíciu môžeme upraviť.

Definícia 3.3.1. *Deterministický konečný automat $A = (K, \Sigma, \delta, q_0, F)$ nazveme* rovnomerne využívaný *na jazyku $L \subseteq L(A)$ s parametrom $k \in \mathbb{N}$ ak existuje $K_0 \subseteq K$ taká, že*

$$\forall w \in L \quad \max_{p \in K - K_0}(\#[p,w]) - \min_{q \in K - K_0}(\#[q,w]) \leq k$$

a $\forall w \in L \quad \max_{p \in K_0}(\#[p,w]) \leq k.$

Podmienka, aby stavy z množiny K_0 boli využívané menej ako k-krát je dôležitá kvôli našej motivácii nájsť definíciu, ktorá by vyhovovala rovnomernému zahrievaniu procesora.

Ak by boli stavy, ktoré neporovnávame s ostatnými používané často, procesor by sa v týchto miestach zahrial.

Poznámka 3.3.1. *Táto definícia nám umožní konštruovať rovnomerne využívané automaty pre jazyk* $\{w\} \cdot L$, *kde* $w = w_1 \dots w_i$ *a* L *má rovnomerne využívaný automat – pridáme nové stavy* $q_{w_1}, \dots, q_{w_i}$, *nový počiatočný stav bude* q_{w_1},$\delta(q_{w_l}, w_l) = q_{w_{l+1}}$ *pre* $l \in \{1, \dots, i-1\}$ *a* $\delta(q_{w_i}, w_i) = q_0$.

Poznámka 3.3.2. *Všimnime si, že ani takáto definícia nám neumožní tvrdiť, že trieda jazykov ku ktorým existuje rovnomerne využívaný automat, je uzavretá na označené zjednotenie, aj keď by to tak na prvý pohľad mohlo vyzerať. Pozrime sa na automat pre označené zjednotenie jazykov* L_1 *a* L_2. *Nech pre tieto jazyky existujú rovnomerne využívané automaty* A_1 *a* A_2. *Bez ujmy na všeobecnosti môžeme predpokladať, že stavy týchto automatov sú rôzne. Automat* A *pre jazyk* $L_1 \cup_c L_2$ *bude mať stavy* $q_0 \cup K_1 \cup K_2$, *kde* q_0 *je nový stav. (Používame štandardnú konštrukciu.) Delta funkcia bude na stavoch z množiny* K_1 *respektíve* K_2 *definovaná rovnako ako v automate* A_1 *respektíve* A_2. *V stave* q_0 *dodefinujeme delta funkciu nasledovne.*

$$\delta(q_0, c) = q_{0_1} \qquad \delta(q_0, x) = \delta_2(q_{0_2}, x) \quad \forall x \in \Sigma_1 \cup \Sigma_2.$$

Z definície delta funkcie vidíme, že $L(A) = L_1 \cup_c L_2$. *Tento automat nie je rovnomerne využívaný, pretože na slovách z jazyka* L_1, *ktoré nepatria do* L_2, *budú stavy z množiny* K_1 *síce rovnomerne využívané, no stavy z množiny* K_2 *ostanú nevyužité. Konkrétnym príkladom môžu byť jazyky* $L_1 = \{a^i \mid i \in \mathbb{N}\}$ *a* $L_2 = \{b^i \mid i \in \mathbb{N}\}$. *Dá sa ukázať, že pre označené zjednotenie týchto dvoch jazykov naozaj neexistuje rovnomerne využívaný automat.*

Ako uvádza poznámka 3.3.2, ani trieda automatov rovnomerne využívaných podľa definície 3.3.1 nebude uzavretá na označené zjednotenie. Ak by sme volili množinu K_0 inú pre každé slovo w, vyriešili by sme tým problém z poznámky 3.3.2. Takto definovaná rovnomerná využívanosť automatu by však už ani zďaleka nevystihovala našu pôvodnú motiváciu, preto si takúto definíciu dovolíme vynechať.

3.3.2 Rovnomerné využívanie na množine slov

Ďalším faktorom ovplyvňujúcim definíciu rovnomerného využívania stavov je jazyk, na ktorom má byť automat rovnomerne využívaný. Ak by sme napríklad povolili, aby rovnomerne využívaný automat na niektorých slovách nevyužíval svoje stavy rovnomerne, stále by sme mali dobrú definíciu tohto pojmu. (Za predpokladu, že slov, ktoré neuvažujeme, bude relatívne málo.) Navyše existujú jazyky, ktoré by vďaka tejto definícii mali rovnomerne využívaný automat.

Na tento parameter definície sa však dá pozrieť aj z iného uhla. Ak by automat využíval svoje stavy nerovnomerne, no po „akceptovaní“ všetkých slov z akceptačného jazyka za sebou by ich využíval rovnomerne, mohli by sme tiež povedať, že je rovnomerne využívaný. V takomto prípade však nebude definovanie tohto pojmu rovnaké ako v predchádzajúcich definíciach. Jazyk, ktorý automat akceptuje môže byť aj nekonečný. Vtedy bude automat využívať svoje stavy „nekonečne veľa krát“.

Možným riešením je zaručiť rovnomerné využívanie stavov na konečných množinách slov tak, aby tieto množiny tvorili rozklad jazyka L. Vhodnou voľbou týchto množín môže byť rozklad podľa dĺžky slova. V praxi však môžeme mať problém s usporiadaním slov. Musíme myslieť na to, že ak dáme slová na vstup v nesprávnom poradí, môže sa procesor prehriať (využívať niektorý svoj stav omnoho viac ako iné) na začiatku a v priebehu spracovávania ďalších slov bude postupne chladnúť. V tomto prípade by bol automat rovnomerne využívaný podľa definície 3.3.2, avšak v skutočnosti by sa zahrial. Táto definícia teda hovorí iba o tom, že pri istom usporiadaní slov môže byť automat rovnomerne využívaný.

Definícia 3.3.2. *Deterministický konečný automat $A = (K, \Sigma, \delta, q_0, F)$ nazveme* rovnomerne využívaný *na jazyku $L \subseteq L(A)$ s parametrom $k \in \mathbb{N}$ ak*

$$\forall n \in \mathbb{N} \quad \max_{p \in K} \left(\sum_{\substack{w \in L \\ |w| = n}} \#[p, w] \right) - \min_{q \in K} \left(\sum_{\substack{w \in L \\ |w| = n}} \#[q, w] \right) \leq k.$$

Poznámka 3.3.3. *Definíciu 3.3.2 môžeme vysloviť v kombinácii s definíciou 3.3.1, a to hneď dvoma spôsobmi. Buď najprv zvolíme pevnú množinu K_0 z definície 3.3.1 alebo povolíme, aby bola množina K_0 rôzna pre každú dĺžku slova n.*

3.3.3 Maximálne využívaný stav

V každej definícii sme doteraz vždy porovnávali dva stavy. (Či už každé dva v prvých definíciách alebo minimálne a maximálne využívané v ďalších.) Ak si však uvedomíme, že v prípade úplne rovnomerného využívania stavov (toho z definície 3.1.1) využívame každý stav $(|w| + 1)/|K|$ krát, môžeme vysloviť definíciu, ktorá neporovnáva dvojicu stavov, ale tvrdí niečo o každom zvlášť. Analogickým dôkazom k dôkazu vety 3.2.1 ukážeme, že nám stačí uvažovať maximálne využívaný stav. Tiež sa zamyslíme nad vzťahom definície s porovnávaním dvoch stavov a definície 3.3.4

Definícia 3.3.3. *Deterministický konečný automat* $A = (K, \Sigma, \delta, q_0, F)$ *nazveme* rovnomerne využívaný *na jazyku* $L \subseteq L(A)$ *s parametrom* $k \in \mathbb{N}$ *ak*

$$\forall w \in L \ \forall p \in K \quad \left| \#[p, w] - \frac{|w|+1}{|K|} \right| \leq k.$$

Definícia 3.3.4. *Deterministický konečný automat* $A = (K, \Sigma, \delta, q_0, F)$ *nazveme* rovnomerne využívaný *na jazyku* $L \subseteq L(A)$ *s parametrom* $k \in \mathbb{N}$ *ak*

$$\forall w \in L \quad \left| \max_{p \in K}(\#[p, w]) - \frac{|w|+1}{|K|} \right| \leq k.$$

Veta 3.3.1. *Ak automat A využíva svoje stavy rovnomerne podľa definície 3.3.3 , tak ich využíva rovnomerne podľa definície 3.3.4 s rovnakým parametrom k.*

Dôkaz. Ak pre každé slovo w a každý stav p platí $|\#[p, w] - (|w| + 1)/|K|| \leq k$ tak pre každé slovo w zrejme platí $| \max_{p \in K}(\#[p, w]) - (|w| + 1)/|K|| \leq k$. $\square$

Poznámka 3.3.4. *Všimnime si automat* $A = (\{q_0, q_1, q_2, q_3\}, \{a, b\}, \delta, q_0, \{q_2\})$*, kde*

$$\begin{aligned} \delta(q_0, a) &= q_1, \\ \delta(q_1, b) &= q_2, \\ \delta(q_2, a) &= q_3, \\ \delta(q_3, b) &= q_0. \end{aligned}$$

Tento automat je rovnomerne využívaný podľa definície 3.3.4 pre $k = 1/4$. *(Najviac využívané stavy sú* q_0*,* q_1 *a* q_2 *– na slovách* $(ab)^{2i+1}$ *každý* $(i + 1)$*-krát.) Automat A nie je rovnomerne využívaný podľa definície 3.3.3 pre* $k = 1/4$*, pretože*

$$\left| \#[q_4, (ab)^{2i+1}] - \frac{4i+3}{4} \right| = \left| i - \frac{4i+3}{4} \right| = \frac{3}{4} > \frac{1}{4}.$$

Preto neplatí obrátená implikácia vo vete 3.3.1.

Poznámka 3.3.5. *Obrátená implikácia v predchádzajúcej vete neplatí pre rovnaký parameter k. Môže nám však záležať len na tom, aby existovali nejaké konštanty* k_1*,* k_2 *tak,* *aby v prípade, že je A rovnomerne vyžívaný podľa definíce 3.3.4 s parametrom k_1, tak je aj rovnomerne využívaný podľa definície 3.3.3 s parametrom k_2. V tom prípade sú definície ekvivalentné – prvú implikáciu sme ukázali v dôkaze vety 3.3.1, opačnú ukážeme nasledovne, sporom.*

Nech je automat A rovnomerne využívaný podľa definície 3.3.4 s parametrom k_1 a zároveň neexistuje k_2 také, aby bol automat A rovnomerne využívaný podľa definície 3.3.3. Uvažujme stav p_1 taký, že pre každé k_2 existuje slovo w_{k_2} také, že

$$\left|\#[p_1, w_{k_2}] - \frac{|w_{k_2}| + 1}{|K|}\right| > k_2.$$

Ak je $\#[p_1, w_{k_2}] > (|w_{k_2}| + 1)/|K|$, tak z toho, že $\max_{p\in K}(\#[p, w_{k_2}]) > \#[p_1, w_{k_2}]$ a pri voľbe $k_2 = k_1$ dostávame, že

$$\left|\max_{p\in K} \#[p, w_{k_2}] - \frac{|w_{k_2}| + 1}{|K|}\right| > k_1,$$

čo je spor s tým, že A je rovnomerne využívaný podľa definície 3.3.4. Preto musí platiť $\#[p_1, w_{k_2}] < (|w_{k_2}| + 1)/|K|$. (Presnejšie $\#[p_1, w_{k_2}] < (|w_{k_2}| + 1)/|K| - k_2$.) Keďže výpočet na slove w_{k_2} má $|w_{k_2}| + 1$ krokov, platí

$$\max_{p\in K} \#[p, w_{k_2}] > \frac{|w_{k_2}| + 1}{|K|} + \frac{k_2}{|K|}.$$

(Inak by $\sum_{p\in K} \#[p, w_{k_2}] \le |w_{k_2}| + 1 - k_2/|K| < |w_{k_2}| + 1$.) Potom ale pre $k_2 = |K|k_1$ platí, že

$$\max_{p\in k} \#[p, w_{k_2}] - \frac{|w_{k_2}| + 1}{|K|} > k_1,$$

čo je spor s predpokladom, že A je rovnomerne využívaný podľa definície 3.3.4.

Všimnime si, že ak je automat rovnomerne využívaný podľa definície 3.2.2, tak pre každé slovo w platí $|\max_{p\in K}(\#[p, w]) - \min_{q\in K}(\#[q, w])| \le k$. Minimálne využívaný stav nemôže byť využitý viac ako $(|w| + 1)/|K|$ krát, pretože stavov je $|K|$ a na slove w sú stavy využité presne $(|w| + 1)$ krát. Keby bol minimálne využívaný stav využitý viac krát, dokopy by boli stavy využité viac ako $|w|+|K|+1$ krát. Touto úvahou sme ukázali, že každý automat, ktorý je rovnomerne využívaný podľa definície 3.2.2, je rovnomerne využívaný aj podľa definície 3.3.4.

Ak sa zamyslíme nad opačnou implikáciou, zistíme, že definície nie sú ekvivalentné. (V zmysle, že automat A je rovnomerne využívaný podľa definície 3.3.4 s parametrom k_A

práve vtedy, keď je s tým istým parametrom rovnomerne využívaný podľa definície 3.2.2.) Automat z poznámky 3.3.4 je rovnomerne využívaný podľa definície 3.3.4 s parametrom $k_A = 1/4$, no podľa definície 3.2.2 až s parametrom $k_A = 1$. Ničmenej, keď nám nezáleží na hodnotách parametrov, ale len na tom, či existujú, sú definície ekvivalentné. Podobnou úvahou ako v poznámke 3.3.5 môžeme zistiť, že

$$\max_{p \in K} \#[p, w] - \min_{q \in K} \#[q, w] \leq k_1 + k_2,$$

kde k_1 je parameter z definície 3.3.4 a k_2 je také prirodzené číslo, že pre všetky slová $w \in L$ platí

$$\min_{q \in K} \#[q, w] \geq \frac{|w| + 1}{|K|} - k_2.$$

Ak by táto nerovnosť pre každé k_2 neplatila, dospeli by sme k sporu s definíciou 3.3.4 pomocou úvahy z poznámky 3.3.5. (Voľbou $k_2 = |K|k_1$.)

3.4 Definície závislé od poradia slov

V tejto kapitole sme uviedli viacero definícií, z ktorých sa niektoré dajú kombinovať. Najvhodnejšou definíciou vzhľadom na čo najľahšiu prácu s ňou pri dokazovaní sa zatiaľ javí definícia 3.3.4, pretože pri nej stačí nájsť iba jednu hodnotu, maximum počtu využitia nejakého stavu na danom slove. Na druhej strane môžeme povedať, že táto definícia je príliš obmedzujúca. Všimnime si napríklad automat $A = (\{q_0, q_1\}, \{a, b\}, \delta, q_0, \{q_1\})$, kde

$$\begin{aligned} \delta(q_0, a) &= q_0, \\ \delta(q_0, b) = \delta(q_1, b) &= q_1, \end{aligned}$$

a $\delta(q_1, a)$ je nedefinovaná. (Ak by sme chceli použiť klasickú definíciu DKA, máme ešte stav q_u a $\delta(q_1, a) = \delta(q_u, a) = \delta(q_u, b) = q_u$.)

Tento automat nebude rovnomerne využívať svoje stavy podľa definície 3.3.4. (Napríklad kvôli slovám $a^i b$, konkrétny dôvod môže čitateľ nájsť v príklade 4.1.5.) Napriek tomu, ak by sme tomuto automatu dávali na vstup slová vo vhodnom poradí, bude svoje stavy využívať rovnomerne. Pod vhodným poradím myslíme napríklad také, v ktorom automat dostane na vstup najprv všetky slová dĺžky 0, potom slová dĺžky 1, a tak ďalej. Automat A je teda rovnomerne využívaný podľa definície 3.3.2.

Definícia 3.3.2 sa dá poupraviť podobne ako definícia 3.2.2 na 3.3.4.

Definícia 3.4.1. *Deterministický konečný automat $A = (K, \Sigma, \delta, q_0, F)$ nazveme* rovnomerne využívaný *na jazyku $L \subseteq L(A)$ s parametrom $k \in \mathbb{N}$ ak*

$$\forall n \in \mathbb{N} \quad \left| \max_{p \in K} \left(\sum_{\substack{|w|=n \\ w \in L}} \#[p, w] \right) - \frac{(n+1) \cdot |L \cap \Sigma^n|}{|K|} \right| \leq k.$$

Výraz $|L \cap \Sigma^n|$ udáva počet všetkých slov jazyka L dĺžky n. Vynásobením $(n + 1)$ dostávame počet všetkých stavov využitých vo výpočtoch na týchto slovách. Po predelení počtom stavov dostaneme počet využití jedného stavu na týchto slovách, ak by sa každý stav použil rovnako často. Ľahko nahliadneme, že rozdiel v podmienke bude vždy nezáporný. (Rovnako ako v definícii 3.3.4.) Môžeme si preto dovoliť písať podmienku bez absolútnej hodnoty.

Môžeme si všimnúť, že ak je jeden zo stavov využívaný častejšie na každom slove dĺžky n (aj keď len o konštantu) a ak takýchto slov je viac ako n, nebude automat podľa tejto definície rovnomerne využívať svoje stavy. Napriek tomu by takýto jazyk bol rovnomerne využívaný podľa definície 3.3.4. Vyslovíme preto ešte slabšiu definíciu, ktorá tento problém vyrieši. Je však dôležité pamätať na našu motiváciu. Procesor, ktorý sa bude zahrievať tak ako povoľuje definícia 3.4.2 sa môže na niektorých častiach zahrievať viac ako na iných, no stále len v závislosti od počtu slov dĺžky n. (A teda počtu vykonaných výpočtov.)

Definícia 3.4.2. *Deterministický konečný automat $A = (K, \Sigma, \delta, q_0, F)$ nazveme* rovnomerne využívaný *na jazyku $L \subseteq L(A)$ s parametrom $k \in \mathbb{N}$ ak*

$$\forall n \in \mathbb{N} \quad \left| \max_{p \in K} \left(\sum_{\substack{|w|=n \\ w \in L}} \#[p, w] \right) - \frac{(n+1) \cdot |L \cap \Sigma^n|}{|K|} \right| \leq k|L \cap \Sigma^n|.$$

Podmienku z definície môžeme prepísať do ekvivalentného tvaru

$$\forall n \in \mathbb{N} \quad \frac{\max_{p \in K} \left(\sum_{\substack{|w|=n \\ w \in L}} \#[p, w] \right)}{|L \cap \Sigma^n|} - \frac{n+1}{|K|} < k,$$

z ktorého vidíme, že definícia je veľmi podobná definícii 3.3.4 – rozdielom je, že si všíma priemerné využitie maximálne využívaného stavu na slovách dĺžky n.

3.5 Výber vhodnej definície

Mohli by sme uvažovať aj o takej zmene definície 3.3.4, v ktorej si nebudeme všímať niektoré stavy automatu. (Podobne ako v definícii 3.3.1.) Problémom je však určiť vhodnú množinu stavov, ktoré si nebudeme všímať. Ak ich zvolíme príliš veľa, definícia stratí svoj význam a automat nebudeme môcť považovať za rovnomerne využívaný. (Ak by 12 stavový automat využíval rovnomerne dva svoje stavy, pravdepodobne by sme už nemáme právo tvrdiť, že využíva svoje stavy rovnomerne.) O jednej z prijateľných volieb množiny stavov, ktoré si nebudeme všímať, hovoria nasledujúce dve definície rovnomerného využívania – prvá definícia je analógiou definície 3.3.4, druhá analógiou 3.4.2.

Definícia 3.5.1. *Deterministický konečný automat* $A = (K, \Sigma, \delta, q_0, F)$ *nazveme* rovnomerne využívaný *na jazyku* $L \subseteq L(A)$ *s parametrom* $k \in \mathbb{N}$ *ak*

$$\forall w \in L \quad \left| \max_{p \in K - K_0} (\#[p, w]) - \frac{|w| + 1}{|K - K_0|} \right| \leq k,$$

kde $K_0 = \{q \in K \mid \exists i \in \mathbb{N}\ \forall w \in L : \#[q, w] < i\}$.

Definícia 3.5.2. *Deterministický konečný automat* $A = (K, \Sigma, \delta, q_0, F)$ *nazveme* rovnomerne využívaný *na jazyku* $L \subseteq L(A)$ *s parametrom* $k \in \mathbb{N}$ *ak*

$$\forall n \in \mathbb{N} \quad \left| \max_{p \in K - K_0} \left(\sum_{\substack{|w|=n \\ w \in L}} \#[p, w] \right) - \frac{(n+1) \cdot |L \cap \Sigma^n|}{|K - K_0|} \right| \leq k|L \cap \Sigma^n|,$$

kde $K_0 = \{q \in K \mid \exists i \in \mathbb{N}\ \forall w \in L : \#[q, w] < i\}$.

Tieto dve definície majú svoje výhody, ale aj nevýhody. Medzi hlavné nevýhody patrí odbočenie od pôvodnej motivácie. Ak by sa procesor zahrieval len na malej časti, už sa nebude v pravom slova zmysle rovnomerne zahrievať. Napriek tomu v teórii automatov takéto definíce majú zmysel, konštrukcie automatov, ktoré sú rovnomerne využívané, sa môžu zjednodušiť. Rozhodli sme sa takéto definície v ďalšom texte neuvažovať, aj keď by takáto možnosť zmenila triedu jazykov, pre ktoré existujú rovnomerne využívané automaty ako aj niektoré vlastnosti triedy rovnomerne využívaných automatov. V niektorých prípadoch upozorníme na skutočnosť, že táto definícia ma iné vlastnosti ako definícia 3.3.4, tieto tvrdenia však budú len informatívne a bez dôkazov.

V ďalšom texte budeme hovoriť, že automat je rovnomerne využívaný podľa nejakej definície, ak existuje konštanta k z definície taká, aby platila podmienka z definície. Navyše budeme uvažovať rovnomerne využívanie stavov na akceptačnom jazyku. (Budeme voliť $L = L(A)$.) Budeme uvažovať definície 3.3.4 a 3.4.2. Tieto definície sme vybrali ako reprezentačné, prvá z nich zaručuje, že stavy budú rovnomerne využívané na všetkých slovách. Druhá povoľuje občasné využívanie niektorých stavov častejšie ako iných, no tieto musia byť využívané menej pre iné slová danej dĺžky. Ľahko nahliadneme, že každý automat, ktorý je rovnomerne využívaný podľa definície 3.3.4, bude rovnomerne využívaný aj podľa definície 3.4.2.

Kapitola 4

Trieda rovnomerných jazykov

V predchádzajúcej kapitole sme zadefinovali pojem rovnomerne využívaného automatu. V ďalšom texte budeme pracovať s vybranými definíciami 3.3.4 a 3.4.2. Zaujíma nás, pre aké jazyky existujú rovnomerne využívané automaty.

Najskôr ukážeme vzťah dvoch automatov akceptujúcich ten istý jazyk vzhľadom na rovnomerné využívanie stavov, vyslovíme uzáverové vlastnosti triedy jazykov prislúchajúcich rovnomerne využívaným automatom a zaradíme túto triedu do Chomského hierarchie. (Budeme samostatne uvažovať obe triedy prislúchajúce k týmto definíciam.)

4.1 Minimálny automat

V ďalšom texte budeme pracovať s minimálnym automatom pre daný regulárny jazyk. Použijeme Myhill-Nerodovu vetu, ktorú uvedieme bez dôkazu. Čitateľ ho však môže nájsť napríklad v [RF08].

Veta 4.1.1 (Myhill-Nerode). *Majme jazyk $L \subseteq \Sigma^*$. Nasledujúce tvrdenia sú ekvivalentné.*

1. *L je regulárny jazyk.*
2. *L je zjednotením niektorých tried ekvivalencie nejakej sprava invariantnej relácie ekvivalencie konečného indexu.*
3. *Relácia R_L definovaná $u \, R_L \, v \iff (\forall x; ux \in L \iff vx \in L)$ je relácia ekvivalencie konečného indexu.*

Podľa Myhill-Nerodovej vety existuje *minimálny automat* – deterministický konečný automat pre daný jazyk L, ktorý má najmenší možný počet stavov. Ten je zostrojený tak, aby každej triede rozkladu relácie R_L z tretej vlastnosti Myhill-Nerodovej vety prislúchal práve jeden zo stavov minimálneho automatu. (Ak je slovo u v nejakej triede rozkladu, automat bude po prečítaní u v stave prislúchajúcom tejto triede.) Zaujíma nás vzťah medzi minimálnym automatom a existenciou rovnomerne využívaného automatu.

Keďže každému stavu minimálneho DKA prislúcha jedna trieda rozkladu, iné DKA pre daný jazyk môžeme zostrojiť z minimálneho tak, že „rozdelíme“ stav minimálneho automatu na viacero stavov. Všimnime si, že stavy prislúchajúce rôznym triedam rozkladu nemôžeme spájať, pretože ak bude automat po prečítaní slov u a v v rovnakom stave a existuje x také, že $ux \in L$ a $vx \notin L$, automat už nebude akceptovať jazyk L.

Pred tým, ako budeme pokračovať, vyslovíme jednu pomocnú vetu. Tá hovorí, že nemôžu existovať stavy, ktoré predchádzajú danému stavu pri každom výpočte rovnako často. Pod tým, že stav q predchádza stav p vo výpočte na slove w budeme rozumieť to, že automat jedným krokom výpočtu prejde zo stavu q do stavu p.

Veta 4.1.2. *Nech $i \in \mathbb{N}$ a DKA $A = (K, \Sigma, \delta, q_0, F)$. Nech existuje stav $p \in K$ taký, že pre všetky $k \in \mathbb{N}$ existuje slovo $w \in L$ také, že $\#[p, w] > k$. Potom neexistujú stavy $s_0, \dots, s_i \in K$ také, že vo výpočte na každom slove w predchádzajú stav p rovnako často.*

Dôkaz. Nech s je jeden zo stavov $s_0, \dots, s_i$ taký, že na pevne danom slove $w = w_0 w_1 \dots w_n$ predchádza druhýkrát p ako prvý z týchto stavov. Nech je to po j_2 krokoch, pričom prvý krát s predchádzal p po j_1 krokoch. Potom pumpovaním časti slova $w_{j_1} \dots w_{j_2}$ dosiahneme to, že stav s bude predchádzať p rádovo viac krát ako ostatné zo stavov $p_0, \dots, p_i$. □

Veta 4.1.3. *Nech DKA A je minimálny automat pre jazyk $L = L(A)$. Pre jazyk L existuje rovnomerne využívaný automat práve vtedy, keď je A rovnomerne využívaný.*

Dôkaz. Tvrdenie platí len pre definíciu 3.3.4.

Ukážeme, že ak A nie je rovnomerne využívaný, žiaden iný automat pre jazyk L nebude rovnomerne využívaný. To, že A nie je rovnomerne využívaný znamená, že pre všetky $k \in \mathbb{N}$ existuje slovo w také, že

$$\left| \max_{p \in K}(\#[p, w]) - \frac{|w| + 1}{|K|} \right| > k.$$

To znamená, že rozdiel $\max_{p \in K}(\#[p, w]) - (|w| + 1)/|K|$ rastie s zväčšujúcou sa veľkosťou slova w. (Výraz $\max_{p \in K}(\#[p, w])$ nemôže byť menší ako $(|w| + 1)/|K|$, inak by mal akceptačný výpočet menej ako $|w| + 1$ krokov, čo je pri deterministických konečných automatoch nemožné.) Nech p je stav, ktorý je maximálne využívaný. Môžeme písať

$$\#[p,w] - \frac{|w|+1}{|K|} > k. \tag{4.1}$$

Pozrime sa teraz na automat B taký, že $L(B) = L$. Ten musel vzniknúť rozdelením niektorých stavov automatu A. (Tak, že jednej triede rozkladu relácie R_L bude prislúchať niekoľko stavov automatu B.) Ak stav p, ktorý je maximálne využívaný v automate A, rozdelíme na stavy $p_0, \ldots, p_i$, môžeme v najlepšom prípade rozdeliť počet využití stavu p rovnomerne medzi stavy $p_0, \ldots, p_i$. (Ak by sme využitie stavu p rozdelili rovnomerne medzi niektoré stavy a ostatné by sme využívali len konštantne veľa krát, boli by pre nás tieto málo využívané stavy skôr príťažou – iba by zmenšovali podiel $(|w|+1)/|K_B|$.) Problémom v tomto prípade bude to, že budeme potrebovať rozdeliť aj všetky ostatné stavy na $i+1$ ďalších – aby sme si v stave pamätali, do ktorého zo stavov $p_0, \ldots, p_i$ máme ísť. Tento údaj by sme si v stave nemuseli pamätať jedine vtedy, keby sme vedeli, že na každom slove automat A prechádza z $i+1$ rôznych stavov do stavu p rovnomerne. To však nemôže platiť. (Veta 4.1.2.) Touto úvahou sme nahliadli, že ak chceme $\max_{p \in K}(\#[p,w])$ znížiť $i+1$-krát, potrebujeme na to automat s $|K|(i+1)$ stavmi. Potom ale podmienka z definície nie je splnená pre žiadne k_B, pretože

$$\frac{\#[p,w]}{i+1} - \frac{|w|+1}{|K|(i+1)} > k_B,$$

platí z 4.1 pre $k = k_B \cdot (i+1)$.

Ukázali sme, že ak pre L existuje rovnomerne využívaný automat, tak minimálny automat A je rovnomerne využívaný. Obrátená implikácia platí triviálne, pretože ak minimálny automat A je rovnomerne využívaný, tak existuje rovnomerne využívaný automat. (Práve ten minimálny.)

□

Poznámka 4.1.1. *Predchádzajúca veta nám umožňuje nazývať jazyky* rovnomerné *podľa definície 3.3.4, pretože vieme jednoznačne povedať, kedy je jazyk rovnomerný – ak je minimálny automat pre tento jazyk rovnomerne využívaný.*

Ukážeme ešte, že automat pre nejaký jazyk nemusí byť rovnomerne využívaný napriek tomu, že minimálny automat pre tento jazyk je rovnomerne využívaný. (A teda nie každý automat pre rovnomerný jazyk je rovnomerne využívaný.)

Príklad 4.1.1. *Jazyk $L = \{a^i \mid i \in \mathbb{N}\}$ je rovnomerne využívaný. (Minimálny automat pre tento jazyk má len jeden stav.) Všimnime si automat $A = (\{q_0, q_1\}, \{a\}, \delta, q_0, \{q_1\})$, kde $\delta(q_0, a) = \delta(q_1, a) = q_1$. Stav q_0 sa využije iba raz, preto takýto automat nie je rovnomerne využívaný.*

Poznámka 4.1.2. *Ak by sme uvažovali rovnomernú využívanosť podľa definície 3.5.1, bol by takto upravený automat tiež rovnomerne využívaný, pretože stavy, ktoré sú využívané konštantne veľa krát vo výpočte na každom slove patria do množiny K_0.*

Ukážeme ešte, že veta 4.1.3 neplatí pre definíciu 3.4.2. Na dvoch príkladoch nahliadneme, že existuje jazyk, ktorého minimálny automat je rovnomerne využívaný, ale existuje aj taký automat, ktorý akceptuje daný jazyk a nie je rovnomerne využívaný, a naopak, že existuje jazyk, ktorého minimálny automat nie je rovnomerne využívaný, ale existuje k nemu aj automat ktorý je rovnomerne využívaný. (Stále uvažujeme rovnomerné využívanie vzhľadom na definíciu 3.4.2.)

Príklad 4.1.2. *Automat $A = (\{q_0, q_1, q_2, q_3\}, \{a, b\}, \delta_A, q_0, \{q_0, q_1, q_2, q_3\})$, kde*

$$\begin{aligned}
\delta_A(q_0, a) &= q_0, \\
\delta_A(q_0, b) = \delta_A(q_3, b) &= q_1, \\
\delta_A(q_1, b) &= q_2, \\
\delta_A(q_2, b) &= q_3,
\end{aligned}$$

akceptuje jazyk $L(A) = L = \{a^i b^j \mid i, j \in \mathbb{N}\}$. Minimálny automat pre tento jazyk je rovnomerne využívaný podľa definície 3.4.2, on sám však nie, pretože

$$\begin{aligned}
\max_{p \in K}\left(\sum_{\substack{|w|=n \\ w \in L}} \#[p, w]\right) - \frac{(n+1)\cdot|L \cap \Sigma^n|}{|K|} &= \sum_{\substack{|w|=n \\ w \in L}} \#[q_0, w] - \frac{(n+1)^2}{4} \\
&= \frac{(n+1)(n+2)}{2} - \frac{(n+1)^2}{4} = \frac{n^2+4n+3}{4},
\end{aligned}$$

a tak neexistuje žiadna konštanta k taká, aby $k(n+1)$ ohraničil tento výraz.

Poznámka 4.1.3. *Tento automat nebude rovnomerne využívaný ani podľa definície 3.5.2, pretože žiaden stav nie je využívaný konštante, a tak $K_0 = \emptyset$.*

Príklad 4.1.3. *Uvažujme dva automaty, ktoré akceptujú slová tvaru $w_1w_2\dots w_{2l-1}w_{2l}$, kde $w_{2i-1}w_{2i}$ je buď ab alebo ba pre všetky $i \in \{1,\dots,l\}$. Prvý z automatov (ten minimálny) označíme $A = (\{q_0,q_1,q_2\},\{a,b\},\delta_A,q_0,\{q_0\})$, kde*

$$\begin{aligned}\delta_A(q_0,a) &= q_1,\\ \delta_A(q_1,b) &= q_0,\\ \delta_A(q_0,b) &= q_2,\\ \delta_A(q_2,a) &= q_0.\end{aligned}$$

Druhý automat $B = \{q_0,q_1,q_2,q_3\},\{a,b\},\delta_B,q_0,\{q_0,q_3\})$, kde

$$\begin{aligned}\delta_B(q_0,a) = \delta_B(q_2,a) &= q_1,\\ \delta_B(q_1,b) = &= q_0,\\ \delta_B(q_0,b) = \delta_B(q_2,b) &= q_2,\\ \delta_B(q_2,a) &= q_3,\end{aligned}$$

vznikol z minimálneho automatu miernou úpravou δ-funkcie a pridaním nového stavu. Počet slov nepárnej dĺžky patriacich do jazyka $L = L(A) = L(B)$ je 0. Počet slov dĺžky $2j$ je 2^j. (Každú dvojica $w_{2i-1}w_{2i}$ môže byť buď ab alebo ba.) Chceme nájsť konštanty k_A a k_B tak, aby platili nerovnosti

$$\max_{p\in K_A}\left(\sum_{\substack{|w|=2j\\ w\in L}} \#[p,w]\right) - \frac{(2j+1)2^j}{3} \le k_A 2^j,$$
$$\max_{p\in K_B}\left(\sum_{\substack{|w|=2j\\ w\in L}} \#[p,w]\right) - \frac{(2j+1)2^j}{4} \le k_B 2^j,$$

V automate A bude $\max_{p\in K_A}\left(\sum_{|w|=2j}\#[p,w]\right) = \sum_{|w|=2j}\#[q_0,w]$, pretože stav q_0 je *na slove dĺžky $2j$ využitý $j+1$ krát, čo je viac ako zvyšné dva stavy. Potom*

$$\left(\sum_{\substack{|w|=2j\\ w\in L}} \#[q_0,w]\right) - \frac{(2j+1)2^j}{3} = (j+1)2^j - \frac{(2j+1)2^j}{3} = \frac{(j+2)2^j}{3}$$

a žiadna konštanta k_A neexistuje. Automat A nie je rovnomerne využívaný podľa definície 3.4.2

V automate B je situácia zložitejšia. Uvažujme slovo dĺžky $2j$ také, že práve p dvojíc písmen $w_{2i-1}w_{2i}$ je tvaru ab a zvyšných $j-p$ je tvaru ba. Potom je stav q_0 na takomto slove využitý $p+1$-krát (vždy na konci dvojice ab a úplne na začiatku výpočtu), stav q_1 je využitý p-krát (vždy v strede dvojice ab), stav q_2 je využitý $j-p$ krát (vždy v strede dvojice ba) a stav q_3 je využitý $j-p$ krát – na konci dvojice ba. Slov dĺžky $2j$, v ktorých p dvojíc $w_{2i-1}w_{2i}$ je tvaru ab je dokopy $\binom{j}{p}$. Preto

$$\sum_{\substack{|w|=2j\\ w\in L}} \#[q_0,w] = \sum_{p=0}^{j} \binom{j}{p}(p+1) = 2^j + j2^{j-1}.$$

Zároveň ľahko overíme, že najvyužívanejším stavom je naozaj q_0. Z uvedeného dostávame

$$\sum_{\substack{|w|=2j\\ w\in L}} \#[q_0,w] - \frac{(2j+1)2^j}{4} = 2^j + j2^{j-1} - \frac{(2j+1)2^j}{4} = \frac{2^{j+2}+2j2^j-(2j+1)2^j}{4} = \frac{3}{4}2^j.$$

Preto existuje konštanta k z definície 3.4.2 a tak sme našli rovnomerne využívaný automat pre jazyk, pre ktorý minimálny automat nie je rovnomerne využívaný.

V nasledujúcom texte zaradíme triedu rovnomerných jazykov do Chomského hierarchie a uvedieme niektoré uzáverové vlastnosti. Označme $\mathcal{B}$ triedu rovnomerných jazykov podľa definície 3.3.4. Kvôli poslednému príkladu budeme uvažovať ešte ďalšie dve triedy jazykov. Trieda $\mathcal{B}_1$ bude označovať triedu jazykov, ktorých minimálny automat je rovnomerne využívaný podľa definície 3.4.2. Trieda $\mathcal{B}_2$ bude označovať triedu jazykov, pre ktoré existuje nejaký rovnomerne využívaný automat podľa definície 3.4.2. Ako sme ukázali v príklade 4.1.3, platí $\mathcal{B}_1 \subset \mathcal{B}_2$. Tiež platí $\mathcal{B}, \mathcal{B}_1, \mathcal{B}_2 \subseteq \mathcal{R}$, pretože jazyky vo všetkých troch triedy určite majú aspoň jeden deterministický konečný automat, a teda sú regulárne. Vieme povedať, že každý konečný jazyk ma minimálny automat, ktorý je rovnomerne využívaný podľa oboch definícií. Na príklade ukážeme, že konečné jazyky sú vlastnou podmnožinou rovnomerných jazykov. Ukážeme tiež, že rovnomerné jazyky sú vlastnou podmnožinou regulárnych jazykov.

Príklad 4.1.4. *Jazyk Σ^* je rovnomerný. Minimálny automat pre tento jazyk je automat $A = (\{q_0\}, \Sigma, \delta, q_0, \{q_0\})$, kde $\delta(q_0, x) = q_0$, pre všetky $x \in \Sigma$. Tento automat je rovnomerne využívaný podľa oboch definícií pre ľubovolné $k \in \mathbb{N}$.*

Príklad 4.1.5. *Jazyk $L = \{a^i b^j \mid i, j \in \mathbb{N}\}$ nie je rovnomerný podľa definície 3.3.4, je však regulárny. (A jeho minimálny je tiež rovnomerne využívaný podľa definície 3.4.2.) Minimálny automat pre tento jazyk je automat[1] $A = (\{q_0, q_1\}, \{a, b\}, \delta, q_0, \{q_0, q_1\})$, kde $\delta(q_0, a) = q_0$, $\delta(q_0.b) = \delta(q_1, b) = q_1$. Tento automat nie je rovnomerne využívaný podľa 3.3.4, pretože pre každé $k \in \mathbb{N}$ a pre slovo a^{4k} platí $\max_{p \in K}(\#[p, a^{4k}]) = \#[q_0, a^{4k}] = 4k$ a $4k - 4k/2 = 2k > k$.*

4.2 Uzáverové vlastnosti rovnomerných jazykov

Budeme skúmať uzavretosť triedy rovnomerných jazykov vzhľadom na niektoré operácie. Tam, kde to bude možné, uvedieme aj uzáverové vlastnosti tried $\mathcal{B}_1$ a $\mathcal{B}_2$.

Veta 4.2.1. *Trieda $\mathcal{B}$ nie je uzavretá na zreťazenie.*

Dôkaz. Jazyky $L_1 = \{a^i \mid i \in N\}$ a $L_2 = \{b^i \mid i \in \mathbb{N}\}$ sú rovnomerné, napriek tomu jazyk $L_1 \cdot L_2 = \{a^i b^j \mid i, j \in \mathbb{N}\}$ nie je. (Príklad 4.1.5.) □

Veta 4.2.2. *Trieda $\mathcal{B}$ nie je uzavretá na zjednotenie.*

Dôkaz. Jazyky $L_1 = \{a^i \mid i \in \mathbb{N}\}$ a $L_2 = \{b^i \mid i \in \mathbb{N}\}$ sú rovnomerné, ale $L_1 \cup L_2$ nie je. Minimálny automat pre $L_1 \cup L_2$ je automat $A = (\{q_0.q_a, q_b\}, \{a, b\}, \delta, q_0, \{q_a, q_b\})$, kde $\delta(q_0, x) = q_x$, $\delta(q_x, x) = q_x$ pre $x \in \{a, b\}$. Tento automat nie je rovnomerne využívaný, pretože pre každé $k \in \mathbb{N}$ a pre slovo a^{3k+2} platí $\max_{p \in K}(\#[p, a^{3k+2}]) = \#[q_a, a^{3k+2}] = 3k+2$ a $3k + 2 - (3k + 3)/3 = 2k + 1 > k$. □

Poznámka 4.2.1. *Môžeme si všimnúť, že trieda $\mathcal{B}$ nie je uzavretá ani na označené zjednotenie, v ktorom sa nerovnomerne využíva počiatočný stav, ktorým rozlišujeme, či slovo patrí do prvého alebo druhého jazyka. Všimnime si tiež, že vo všeobecnosti nám nepomôže*

[1] tento automat nie je úplne DKA, pretože δ funkcia je v skutočnosti relácia, nie funkcia. (Pre niektoré hodnoty je nedefinovaná a automat sa zasekne.) Správne by mal mať automat tri stavy, z toho dva akceptačné a jeden „upratovací“, do ktorého sa dostane stále, keď bola δ funkcia nedefinovaná a z ktorého nemôže odísť.

ani neuvažovanie stavov množiny K_0 podľa definície 3.5.1*, pretože na slovách z prvého jazyka sa nevyužívajú stavy, ktoré pribudli z automatu pre druhý jazyk. Tieto však nemusia byť málo využívané. (Nemusia patriť do množiny K_0.)*

Veta 4.2.3. *Trieda $\mathcal{B}$ nie je uzavretá na komplement*

Dôkaz. Deterministický automat pre jazyk L prerobíme na automat pre jazyk L^C veľmi jednoducho – stačí zameniť akceptačné stavy za neakceptačné. Môže sa zdať, že ak bol pôvodný automat rovnomerne využívaný, automat pre komplement jazyka nema prečo nebyť rovnomerne využívaný. Musíme si však uvedomiť, že definícia rovnomerného využívania na jazyku L hovorí len o slovách z jazyka L a nič ostatných slovách. (Tých z jazyka L^C.) Preto sa môže ľahko stať, že automat pre L^C už nebude rovnomerne využívaný. Minimálny automat pre jazyk $L = \{a\}$ je $A = (\{q_0, q_1, q_2\}, \{a\}, \delta, q_0, \{q_1\})$, kde $\delta(q_0, a) = q_1$ a $\delta(q_1, a) = \delta(q_2, a) = q_2$. Tento automat je rovnomerne využívaný. Ak však budeme uvažovať automat pre L^C (zmeníme množinu akceptačných stavov na $\{q_0, q_2\}$), zistíme, že stav q_2 je na slove dĺžky n využitý $n-1$ krát, a preto jazyk L^C nie je rovnomerný. □

Poznámka 4.2.2. *Dôkazy pre neuzavretosť na zreťazenie, zjednotenie a komplement sa dajú analogicky sformulovať aj pre definíciu 3.3.1.*

Veta 4.2.4. *Trieda $\mathcal{B}$ nie je uzavretá na inverzný homomorfizmus.*

Dôkaz. Uvažujme homomorfizmus h, ktorý zobrazí b na b a a na ε. Jazyk $L = \{b\}$ je rovnomerný, pretože je konečný, ale $h^{-1}(L) = \{a^i b a^j \mid i, j \in \mathbb{N}\}$ už rovnomerný nie je, pretože minimálny automat pre tento jazyk má tri stavy a pri slovách, v ktorých je $i > j$ sa prvý stav využíva viac ako tretí. (Pre veľký rozdiel $i - j$ rastie aj počet využití prvého stavu oproti počtu využití trietieho stavu.) □

Veta 4.2.5. *Trieda $\mathcal{B}$ nie je uzavretá na substitúciu.*

Dôkaz. Jazyk $L = \{aba\}$ je rovnomerný, pretože je konečný. Všimnime si substitúciu φ, ktorá zobrazí znak b na jazyk $\{b\}$ a znak a na jazyk $\{a^i \mid i \in \mathbb{N}\}$. (Ako sme už uviedli v predchádzajúcom texte, jazyky $\{b\}$ a $\{a^i \mid i \in \mathbb{N}\}$ sú rovnomerné.) Jazyk $\varphi(L) = \{a^i b a^j \mid i, j \in \mathbb{N}\}$ už rovnomerný nie je, ako sme ukázali v predchádzajúcom dôkaze. □

Poznámka 4.2.3. *Všimnime si, že dôkaz posledných dvoch viet sa nezmení, ani keď budeme uvažovať triedu $\mathcal{B}_1$. Preto môžeme prehlásiť, že trieda $\mathcal{B}_1$ nie je uzavretá na inverzný homomorfizmus ani na substitúciu.*

Veta 4.2.6. *Triedy $\mathcal{B}$ a $\mathcal{B}_1$ nie sú uzavreté na iteráciu a kladnú iteráciu.*

Dôkaz. Konečný jazyk $L = \{ab, ba\}$ patrí do oboch tried. Jazyk L^* je jazyk z príkladu 4.1.3, kde sme ukázali, že $L^* \notin \mathcal{B}_1$. (Rovnako ukážeme, že $L^+ \notin \mathcal{B}_1$.) Keď si všimneme výpočet minimálneho automatu pre tento jazyk na slove $(ab)^i$, zistíme, že $\#[q_0, (ab)^i] = i + 1$, a preto minimálny automat pre tento jazyk nie je rovnomerne využívaný podľa definície 3.3.4, z čoho vyplýva $L^* \notin \mathcal{B}$. (Pre jazyk L^+ by dôkaz vyzeral analogicky.) □

Kapitola 5

Záver

V práci sme podali rôzne pohľady na rovnomerné využívanie stavov konečných automatov a analyzovali sme viaceré definície. Našou snahou bolo vybrať z definícií tú, ktorá by čo najpresnejšie vystihovala pôvodnú motiváciu za hľadaním rovnomerne využívaného automatu – rovnomerné zahrievanie procesora. Napriek tomu, že sme pri niektorých definíciach boli nadmieru mierni, trieda jazykov, ku ktorým existujú rovnomerne využívané automaty, bola vždy menšia ako regulárne jazyky. Tento výsledok hovorí, že vo všeobecnosti nemusí existovať rovnomerne využívaný automat k deterministickému automatu. Preto sme sa skúmali triedu rovnomerných jazykov a jej uzáverové vlastnosti. Ukázalo sa však, že rovnomerné jazyky nie sú uzavreté ani na také základné operácie, akými sú zreťazenie alebo zjednotenie. Medzi tie pozitívnejšie výsledky práce patrí tvrdenie, že ak pre jazyk existuje rovnomerne využívaný automat, tak aj minimálny automat pre tento jazyk bude rovnomerne využívaný. Samozrejme, ako pri väčšine tvrdení v tejto práci, platnosť tohto tvrdenia je obmedzená len na niektoré definície rovnomerného využívania stavov.

Hlavným problémom pri návrhu definície rovnomerného využívania pre nás bolo dostatočne presne dodržať motiváciu, aby sa nestalo, že definícia už nebude hovoriť nič o rovnomernosti využívania stavov, ako aj fakt, že rovnomerné využívanie stavov v zmysle, ktorý v práci uvažujeme, nebolo doposiaľ skúmané.

Termín rovnomerného využívania stavov deterministických automatov nie je triviálne transformovateľný na nedeterministické automaty. Problémov je hneď viacero, pri nedeterminizme nemáme jednoznačne určený výpočet automatu. Napriek tomu veríme, že výsledky pre nedeterministické automaty by mohli byť o niečo priaznivejšie, hlavne kvôli jednoduchším konštrukciám automatov pre prienik, zjednotenie, ale aj ostatné operácie s jazykmi.

Po zistení, že rovnomerné jazyky sú vlastnou podmnožinou regulárnych jazykov je namieste rozmýšľať o miere rovnomernosti využívania automatov. Táto miera sa dá zaviesť pomocou definícií v tejto práci napríklad tak, že namiesto konštanty k, ktorá ohraničuje istú podmienku rovnomernosti v definícii, budeme hovoriť o funkcii $f(n)$, ktorá ohraničí túto podmienku v závislosti od dĺžky slova. Môžeme potom prehlásiť, že nejaký automat A je rovnomernejšie využívaný ako iný automat B, ak funkcia $f_A(n)$ rastie asymptoticky pomalšie ako funkcia $f_B(n)$.

V práci sme ponechali otvorené otázky uzavretosti rovnomerných jazykov na operácie *prienik*, *homomorfizmus* a *reverz*. Našou hypotézou je neuzavretosť na žiadnu z nich. Všetky spomenuté otázky, od uzáverových vlastností po nedeterminizmus, plánujeme skúmať v ďalšej práci.

Literatúra

[HMU01] John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman. *Introduction to automata theory, languages, and computation.* Addison-Wesley Longman Publishing, second edition, 2001.

[HU69] John E. Hopcroft and Jeffrey D. Ullman. *Formal languages and their relation to Automata.* Addison-Wesley Longman Publishing, 1969.

[RF08] Branislav Rovan and Michal Forišek. Formálne jazyky a automaty (skriptá). `http://foja.dcs.fmph.uniba.sk/~misof/foja-skripta-snapshot.pdf`, 2008.