FI MU

Fakulta informatiky
Masarykova univerzita

Automaty nad nekonečnými slovy

Mojmír Křetínský

Učební text FI MU **verze 1.0**

 prosinec 2018

Obsah

Kapitola 1

Büchiho automaty

1.1 Jazyky nekonečných slov

Nechť Σ je konečná abeceda. Σ^* značí množinu všech slov konečné délky, typicky označovaných $u, v, w, \ldots \in \Sigma^*, w = a_0 a_1 \ldots a_n,$ kde $a_i \in \Sigma$. Slovo nekonečné délky, nazývané též ω–slovo, definujeme jako funkci $\alpha : \mathbb{N}_0 \to \Sigma$. Symbolem Σ^ω značíme množinu všech slov nekonečné délky, typicky označovaných $\alpha, \beta \ldots \in \mathit{\Sigma}^\omega$. Tedy $\alpha(i)$ označuje písmeno na i–té pozici a ω–slovo α můžeme a často též budeme zapisovat jako $\alpha = \alpha(0)\alpha(1)\ldots$. Dále označme $\Sigma^\infty = \Sigma^* \cup \Sigma^\omega$.

Pro přirozená m, n taková, že $m \leq n$ definujeme

$$\begin{aligned}
\alpha[m, n] &= \alpha(m)\alpha(m+1)\ldots\alpha(n) \\
\alpha[m, n) &= \alpha(m)\alpha(m+1)\ldots\alpha(n-1) \\
\alpha[m, \omega] &= \alpha(m)\alpha(m+1)\ldots
\end{aligned}$$

V dalším textu použijeme pro kvantifikátory "existuje nekonečně mnoho n" a "existuje pouze konečně mnoho n" jako zkratky symboly $\exists^\omega n$ resp. $\exists^{<\omega} n$. Formálně definujeme $\exists^\omega n.\phi(n)$ jako zkratku pro $\forall i.\exists n.(n > i \wedge \phi(n))$ a dále $\exists^{<\omega} n$ jako zkratku pro $\exists n.\phi(n) \wedge \exists i \forall j.(j > i \Rightarrow \neg\phi(j))$. Analogicky zápis $\forall^\omega n.\phi(n)$ budeme interpretovat jako zkratku pro $\exists i.\forall n.(n > i \Rightarrow \phi(n))$.

Nechť $X \subseteq \Sigma^*, Y \subseteq \Sigma^\infty$. Pro $x \in X, y \in Y$ definujeme jejich zřetězení jako slovo $x.y = x(0)x(1)\ldots x(n)y(0)y(1)\ldots,$ kde $x(i) = x_i$ (symbol '.' často vynecháváme, tj. místo $x.y$ píšeme stručněji xy).

Dále zřetězení jazyků $X \subseteq \Sigma^*$ a $Y \subseteq \Sigma^\infty$ definujeme jako jazyk

$$X.Y = \{x.y \mid x \in X, y \in Y\}$$

a nekonečnou iteraci (ω–iteraci) jazyka $X \subseteq \Sigma^*$ jako jazyk

$$X^\omega = \{x_0 x_1 \ldots \mid x_i \in X - \{\varepsilon\}\}$$

tj. X^ω je množina všech nekonečných slov, která získáme zřetězením nekonečné posloupnosti neprázdných slov z X. Je-li například $u = a_0 a_1 \ldots a_n, a_i \in \Sigma$ a $X = \{u\}$, pak $X^\omega = \{u^\omega\}$, kde u^ω je nekonečné slovo $a_0 a_1 \ldots a_n a_0 a_1 \ldots a_n a_0 \ldots$, které též můžeme zapsat jako $uu\ldots$. Pro $Y = \{x, y\}$ je $Y^\omega = \{xx\ldots, xy\ldots, yx\ldots, yy\ldots, \ \ldots\}$.

Dále pro $X \subseteq \Sigma^*$ definujme

$$\mathsf{pref}(X) = \{u \in \Sigma^* \mid \exists v.uv \in X\}$$
$$\mathsf{lim}(X) = \{\alpha \in \Sigma^\omega \mid \exists^\omega n.\alpha[0,n] \in X\}\ ,$$

tj. $\alpha \in \mathsf{lim}(X)$, právě když α má nekonečně mnoho prefixů v X. Poznamenejme, že často používanou notací pro $\mathsf{lim}(X)$ jsou též značení $\vec{X}$ či X^δ.

Operátory ω–iterace a lim jsou oba typu $\Sigma^* \to \Sigma^\omega$, a tedy umožňují definovat nekonečná slova pomocí slov konečné délky.

Příklad 1.1. (a) Je-li $U = a^*b$, pak $\mathsf{lim}(U) = \emptyset$.

(b) Je-li $U = (ab)^+$, pak $\mathsf{lim}(U) = (ab)^\omega$.

(c) Je-li $U = (a^*b)^+ = (a+b)^*b$, tj. U je množina všech slov končících symbolem b, pak $\mathsf{lim}(U) = (a^*b)^\omega$ je množinou všech ω–slov obsahujících nekonečně mnoho výskytů b.

1.2 ω–automaty

Přechodový systém A (s návěštími a počátečními stavy), zkráceně LTS (labelled transition system) definujeme jako

$$A = (S, \Sigma, \Delta, S_{in}),$$

kde S je množina stavů, Σ je konečná (vstupní) abeceda, $\Delta \subseteq S \times \Sigma \times S$ je přechodová relace a $S_{in} \subseteq S$ je množina počátečních stavů. Namísto $(s, a, s') \in \Delta$ často píšeme $s \xrightarrow{a}_\Delta s'$, či pouze $s \xrightarrow{a} s'$, pokud je Δ zřejmá z kontextu. Někdy též mluvíme o přechodovém systému *nad* abecedou Σ a v tom případě jej zapisujeme pouze jako $A = (S, \Delta, S_{in})$. Poznamejme, že takto definovaný LTS je obecně nedeterministický, mimo jiné proto, že Δ je relace, nikoliv funkce z $S \times \Sigma$ do S.

Přechodový systém A nazveme *deterministický*, jestliže $card(S_{in}) = 1$ a přechodová relace je funkcí $\Delta : S \times \Sigma \to S$. V případě deterministických LTS lze bez újmy na obecnosti předpokládat, že Δ je totální, tj. definovaná pro všechny argumenty (v opačném případě přidáme nový "záchytný" stav, do nějž vedeme všechny dosud nedefinované přechody; přechody z tohoto nově přidaného stavu vedou opět jen do tohoto stavu).

Je-li $u = a_1a_2\dots a_m$ slovo nad Σ, pak klademe $s \xrightarrow{u} s'$, právě když existují $s_0, s_1, \dots, s_m \in S$ taková, že $s = s_0,\ s_{i-1} \xrightarrow{a_i} s_i\ (0 < i \le m)$ a $s_m = s'$.

Běh LTS. Nechť $A = (S, \Delta, S_{in})$ je LTS nad Σ, $w \in \Sigma^*, w = a_0a_1\dots a_n$ konečné vstupní slovo a $\alpha : \mathbb{N}_0 \to \Sigma$ vstupní ω–slovo. Běh ρ systému A na slově w je konečná posloupnost $s_0, s_1, \dots s_{n+1}$ stavů z S taková, že $\forall i.\ 0 < i \le m : s_i \xrightarrow{a_i} s_{i+1}$. Běh ρ systému A na ω–slově α je definován jako nekonečná posloupnost $\rho : \mathbb{N}_0 \to S$ taková, že $\forall i\ .i \in \mathbb{N}_0 : \rho(i) \xrightarrow{\alpha(i)} \rho(i+1)$. Je-li navíc $\rho(0) \in S_{in}$, resp. $s_0 \in S_{in}$ nazýváme běh iniciálním.

Běh je tedy posloupnost stavů, jimiž může A projít při postupném čtení slova dle pravidel přechodové relace. V dalším textu, pokud nebude řečeno jinak, budeme termínem běh rozumět iniciální běh, ostatní běhy nazveme neiniciální či obecné běhy.

Dále pro libovolné ω–slovo $\rho : \mathbb{N}_0 \to S$ klademe

$$\mathsf{inf}(\rho) = \{s \in S \mid \exists^\omega n.\ \rho(n) = s\},$$
$$\mathsf{occ}(\rho) = \{s \in S \mid \exists n.\ \rho(n) = s\}$$

tedy $\mathsf{inf}(\rho)$ definujeme jako množinu symbolů z S, které se vyskytují v ω–slově ρ nekonečně často, kdežto $\mathsf{occ}(\sigma)$ je množina symbolů z S, které se v σ vyskytují.

ω–automat. Nechť $A = (S, \Delta, S_{in})$ je LTS nad Σ, pak ω–automatem nazveme systém $\mathcal{A} = (A, Acc)$, kde Acc je tzv. akceptační podmínka definující, kdy je ω–slovo akceptováno, a to obvykle na základě existence běhu jistých vlastností na daném slově. Takový běh nazveme úspěšný nebo též akceptující. Poznamenejme, že ω–automat je obecně nedeterministický, a tedy může pro ω–slovo α existovat více vzájemně různých běhů.

Je-li A konečně stavový, pak $\mathcal{A}$ nazveme konečným ω–automatem. Pokud nebude v dalším textu řečeno jinak, budeme mít na mysli pouze konečné automaty (i když některé z dále uváděných výsledků platí i pro automaty nekonečné s nejvýše spočetnou množinu stavů). Řekneme, že ω–automat $\mathcal{A}$ je deterministický, právě když jeho LTS A je deterministický.

Büchiho automat. Nechť $A = (S, \Delta, S_{in})$ je LTS nad Σ. Řekneme, že ω–automat (A, Acc) je Büchiho automatem, právě když Acc je tzv. Büchi akceptační podmínka (BAC):

je dána konečná množina $F \subseteq S$ akceptujích (též finálních) stavů a slovo $\alpha : \mathbb{N}_0 \to \Sigma$ je akceptováno, právě když existuje běh $\rho : \mathbb{N}_0 \to S$ na slově α takový, že $\mathsf{inf}(\rho) \cap F \neq \emptyset$. V tomto případě říkáme, že $\mathcal{A}$ *akceptuje* α.

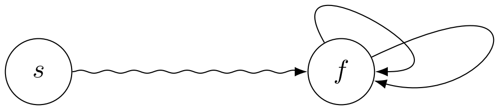
Büchiho podmínku můžeme stručněji zapsat takto:

(BAC) $$\exists \rho.(\rho(0) \in S_{in} \ \wedge\ \forall i.(\rho(i) \overset{\alpha(i)}{\to} \rho(i+1) \in \Delta)\ \wedge\ \mathsf{inf}(\rho) \cap F \neq \emptyset$$

Uvedený Büchiho automat $\mathcal{A}$ značíme (A, F) nebo též $(S, \Sigma, \Delta, S_{in}, F)$. Jazyk rozpoznávaný (též akceptovaný) Büchiho automatem $\mathcal{A} = (A, F)$ je množina

$$L(\mathcal{A}) = \{\alpha \in \Sigma^\omega \mid \mathcal{A} \text{ akceptuje } \alpha\}$$

všech ω–slov akceptovaných tímto automatem a značíme jej též $L(A, F)$. Konečně řekneme, že množina $L \subseteq \Sigma^\omega$ je rozpoznatelná ve smyslu Büchiho (či Büchi–rozpoznatelná), právě když existuje Büchiho automat $\mathcal{A}$ takový, že $L = L(\mathcal{A})$. Automaty $\mathcal{A}, \mathcal{B}$ nazveme ekvivaletní, právě když $L(\mathcal{A}) = L(\mathcal{B})$.



Obrázek 1.1: typický úspěšný běh Büchiho automatu, $s \in S_{in}, f \in F$

Poznámka 1.2. Büchiho akceptační podmínka tedy požaduje, aby se nějaká podmnožina množiny F vyskytovala v úspěšném běhu ρ nekonečně častokrát: jelikož F je konečná, musí existovat nějaký stav $f \in F$, který se v ρ vyskytuje nekonečněkrát. Pokud si představíme přechodový graf Büchiho automatu, pak cesta tímto grafem koresponující úspěšnému běhu začíná v nějakém stavu z S_{in}, dosáhne stavu f a podél nějaké cesty se do f opět (nekonečně častokrát) vrací – viz též obrázek 1.1. $\square$

Podmínku $\mathsf{inf}(\rho) \cap F \neq \emptyset$ (tj. $\forall i \exists j.\, j > i : \rho(j) \in F$) z BAC kladenou na stavy úspěšného běhu ρ můžeme formálně zapsat jako formuli predikátového počtu prvního řádu takto:

$$\bigvee_{q \in F} \forall i \exists j.\, j > i : \rho(j) = q$$

Příklad 1.3. Mějme Büchiho automaty $\mathcal{A} = (\{q_1, q_2\}, \{a, b\}, \Delta, \{q_1\}, \{q_2\})$, kde Δ je dána takto:

$$\begin{array}{ll} q_1 \overset{a}{\to} q_2 & q_1 \overset{b}{\to} q_1 \\ q_2 \overset{a}{\to} q_1 & q_2 \overset{b}{\to} q_1 \end{array}$$

a $\mathcal{B} = (\{q_1, q_2\}, \{a, b\}, \Delta, \{q_1\}, \{q_2\})$, kde Δ je dána takto:

$$\begin{array}{ll} q_1 \overset{a}{\to} q_2 & \\ q_2 \overset{a}{\to} q_1 & q_2 \overset{b}{\to} q_1 \end{array}$$

Obrázek 1.2: Automaty $\mathcal{A}$ (vlevo) a $\mathcal{B}$ (vpravo)

Automat $\mathcal{A}$ je deterministický a s totální přechodovou funkcí (resp. jeho přechodový systém má tyto vlastnosti); rovněž $\mathcal{B}$ je deterministický, ale jeho přechodová funkce je parciální (sestrojte ekvivalentní automat s totální Δ).

Automat $\mathcal{A}$ akceptuje množinu všech ω–slov takových, která obsahují nekonečně mnoho výskytů symbolu a, kdežto $L(\mathcal{B})$ je tvořen všemi slovy, u nichž je každý sudý přechod pod písmenem a.

Úkol: Sestrojte Büchiho automat nad $\{a, b\}$ akceptující množinu právě všech takových slov, v nichž se symboly a a b se vykytují nekonečně mnohokrát, přičemž mezi každými dvěma výskyty symbolu a je sudý počet výskytů symbolu b.

Pro konečné automaty (FA) pracující nad slovy konečné délky platí, že nedeterminismus nezvětšuje vyjadřovací sílu, tj. ke každému nedeterministickému konečnému automatu (NFA) existuje ekvivaletní deterministický konečný automat (DFA). Zabývejme se nyní otázkou, zda platí tento vztah u Büchiho automatů.

Příklad 1.4. Nechť je dán Büchiho automat $\mathcal{A}_1 = (\{s_1, s_2\}, \{a, b\}, \Delta_1, \{s_1\}, \{s_1\})$, kde Δ_1 je dána takto:

$$\begin{array}{ll} s_1 \stackrel{a}{\to} s_1 & s_1 \stackrel{b}{\to} s_2 \\ s_2 \stackrel{a}{\to} s_1 & s_2 \stackrel{b}{\to} s_2 \end{array}$$

a Büchiho automat $\mathcal{A}_2 = (\{s_1, s_2\}, \{a, b\}, \Delta_2, \{s_1\}, \{s_2\})$, kde Δ_2 je dána takto:

$$\begin{array}{lll} s_1 \stackrel{a}{\to} s_1 & s_1 \stackrel{b}{\to} s_1 & s_1 \stackrel{b}{\to} s_2 \\ & s_2 \stackrel{b}{\to} s_2 & \end{array}$$

Obrázek 1.3: Automaty $\mathcal{A}_1$ (vlevo) a $\mathcal{A}_2$ (vpravo)

Automat $\mathcal{A}$ z příkladu 1.3 je ekvivalentní s výše uvedeným $\mathcal{A}_1$ – oba akceptují množinu všech ω–slov obsahujících nekonečně mnoho výskytů symbolu a (zde však, na rozdíl od $\mathcal{A}$, všechny a–přechody vedou do akceptujícího stavu s_1 a současně všechny přechody do s_1 jsou pod symbolem a).

Automat $\mathcal{A}_2$ akceptuje množinu všech ω–slov, která je komplementem $L(\mathcal{A}_1)$. $\mathcal{A}_2$ je nedeterministický – uhodne místo ve vstupním slově, za nímž již nejsou žádné výskyty a. Taková pozice ve slově musí existovat, neboť libovolné vstupní slovo, které má být akceptováno, má jen konečně mnoho výskytů a. Automat pak kontroluje správnost svého hádání tím, že může číst pouze symboly b; z s_2 totiž nevede žádný a–přechod a $\mathcal{A}_2$ by nebyl schopen pokračovat ve čtení vstupního slova, a tedy by jej nemohl akceptovat. □

Poznámka 1.5. Jak již bylo uvedeno, $\mathcal{A}$ z příkladu 1.3 a $\mathcal{A}_1$ z příkladu 1.4 jsou ekvivaletní. Jelikož v úspěšném běhu se musí symbol a vyskytovat nekonečně mnohokrát, není nutno, aby každý výskyt a byl zaznamenán průchodem přes akceptující stav – $\mathcal{A}$ prochází přes akceptující stavy jen při každém lichém výskytu symbolu a.

Ve výše uvedeném příkladu 1.4 platí $L(\mathcal{A}_2) = \{a, b\}^* - L(\mathcal{A}_1)$. Povšimnněme si, že $\mathcal{A}_1$ je deterministický, ale $\mathcal{A}_2$ je nedeterministický. Ukážeme, že nedeterminismus je v tomto případě nevyhnutelný v tom smyslu, že *neexistuje* žádný deterministický Büchiho automat, který by rozpoznával komplement jazyka $L(\mathcal{A}_1)$. □

Věta 1.6. *Jazyk $L \subseteq \Sigma^\omega$ je rozpoznatelný deterministickým Büchiho automatem, právě když existuje regulární jazyk $U \subseteq \Sigma^*$ takový, že $L = \lim(U)$.*

Důkaz. $\Longleftarrow$: Nechť U je regulární jazyk nad Σ. Pak existuje DFA $\mathcal{A} = (S, \Sigma, \Delta, q_0, F)$ takový, že $L(\mathcal{A}) = U$. Pokud $\mathcal{A}$ interpretujeme jako Büchiho automat, pak $L(\mathcal{A}) = \lim(U)$ a protože přechodový systém automatu $\mathcal{A}$ se touto interpretací nezměnil, máme Büchiho automat deterministický.

$\Longrightarrow$: Nechť L je rozpoznáván *deterministickým* Büchiho automatem $\mathcal{A}$ s množinou akceptujích stavů F. Pokud F interpretujeme jako množinu koncových stavů DFA s týmž přechodovým systémem jako má $\mathcal{A}$, pak tento DFA akceptuje jistý regulární jazyk, řekněme U. Opět se snadno nahlédne, že $L = \mathsf{lim}(U)$. $\square$

Důsledek 1.7. *Existují jazyky rozpoznatelné nedeterministickými Büchiho automaty, které nejsou rozpoznatelné žádnými deterministickými Büchiho automaty.*

Důkaz. Ukážeme, že jazyk $\bar{L} = L(\mathcal{A}_2)$ z příkladu 1.4 není tvaru $\mathsf{lim}(U)$ pro žádný regulární jazyk U. Připomeňme, že $\bar{L}$ je množinou všech ω–slov nad $\Sigma = \{a, b\}^*$ obsahujících nekonečně mnoho výskytů symbolu b, ale pouze konečně mnoho výskytů symbolu a.

Sporem. Předpokládejme, že $\bar{L}$ je rozpoznatelný deterministickým Büchiho automatem. Pak dle Věty 1.6 existuje regulární $U \subseteq \Sigma^*$ nad takový, že $\bar{L} = \mathsf{lim}(U)$. Jelikož $b^\omega \in \bar{L}$, pak musí existovat jeho nějaký konečný prefix $b^{n_1} \in U$. Odtud a z definice jazyka $\bar{L}$ plyne, že $b^{n_1}ab^\omega \in \bar{L}$, a tedy opět musí existovat nějaký konečný prefix $b^{n_1}ab^{n_2} \in U$. Tedy opět z definice $\bar{L}$ plyne, že $b^{n_1}ab^{n_2}ab^\omega \in \bar{L}$, a tudíž musí existovat další konečný prefix $b^{n_1}ab^{n_2}ab^{n_3} \in U$. Tímto postupem obdržíme nekonečnou posloupnost slov z U: $\{b^{n_1}, b^{n_1}ab^{n_2}, b^{n_1}ab^{n_2}ab^{n_3}, \dots\ \} \subseteq U$. Z definice operátoru lim plyne, že i ω–slovo $\beta = b^{n_1}ab^{n_2}ab^{n_3}a \dots ab^{n_i} \dots \in \mathsf{lim}(U)$. Avšak β má nekonečně mnoho výskytů symbolu a, takže jistě nepatří do $\bar{L}$, což je ale spor s předpokladem, že $\bar{L} = \mathsf{lim}(U)$. $\square$

Ukázali jsme tedy, že třída jazyků rozpoznávaných deterministickými Büchiho automaty je vlastní podtřídou jazyků rozpoznávaných nedeterministickými Büchiho automaty.

1.3 Uzávěrové vlastnosti Büchi–rozpoznatelných jazyků

V této části se budeme zabývat studiem některých uzávěrových vlastností Büchi–rozpoznatelných jazyků. Všechny uvedené důkazy budou konstruktivní a uvedené techniky lze tedy použít pro návrh těchto automatů.

Věta 1.8. *Jsou-li $L_1, L_2 \subseteq \Sigma^\omega$ Büchi–rozpoznatelné jazyky, pak $L_1 \cup L_2$ a $L_1 \cap L_2$ jsou Büchi–rozpoznatelné.*

Důkaz. Mějme Büchiho automaty $\mathcal{A}_i = (S_i, \Sigma, \Delta_i, S^i_{in}, F_i)$ takové, že $L_i = L(\mathcal{A}_i),\ i = 1, 2$.

(a) Uzavřenost vůči sjednocení lze dokázat stejným postupem jako v případě nedeterministických konečných automatů: bez újmy na obecnosti lze předpokládat, že množiny stavů těchto automatů jsou disjunktní. Pak $L_1 \cup L_2$ je rozpoznáván automatem $\mathcal{A} = (S_1 \cup S_2, \Sigma, \Delta_1 \cup \Delta_2, S^1_{in} \cup S^2_{in}, F_1 \cup F_2)$. Počet stavů automatu $\mathcal{A}$ je tedy roven součtu počtu stavů automatů $\mathcal{A}_i$.

(b) Uzavřenost vůči průniku se v případě konečných automatů dokazuje konstrukcí automatu realizujícího tzv. synchronní paralelní spojení daných automatů – stavový prostor automatu rozpoznávajího průnik je kartézským součinem stavových prostorů výchozích automatů a odpovídajícím způsobem jsou definovány i ostatní komponenty, například množina koncových stavů je opět kartézským součinem původních množin koncových stavů.

Pro případ nekonečných slov však musíme uvedenou konstrukci poněkud upravit. Nekonečné slovo α má být akceptováno, jestliže každá z obou synchronních kopií $\mathcal{A}_i,\ i = 1, 2$ prochází při běhu na slově α akceptujícími stavy nekonečně častokrát. Bohužel však není garantováno, že při takovém běhu by akceptující stavy obou automatů byly navštěvovány současně. Například jeden automat by procházel akceptujícími stavy po přečtení $\alpha(0), \alpha(2), \ldots$, kdežto druhý po přečtení $\alpha(1), \alpha(3), \ldots$. Tedy za množinu akceptujících stavů průnikového automatu nelze vzít $F_1 \times F_2$.

Klíčovým pozorováním je fakt (viz poznámka 1.5), že nemusíme zaznamenávat každý výskyt, kdy každá z kopií navštěvuje své akceptující stavy. Stačí, aby každá z těchto kopií zanamenávala jen nekonečnou podposloupnost své původní posloupnosti akceptujících stavů. Jinak řečeno, začneme s tím, že u první kopie (komponenty) budeme čekat na průchod akceptujícím stavem a jakmile toto nastane, přeneseme svoji pozornost na očekávání výskytu akceptujícího stavu u druhé kopie; vejde-li tato do svého akceptujícího stavu, přepneme zpět na pozorování výskytu akceptujícího stavu u první kopie atd. Je vidět, že budeme přepínat mezi kopiemi nekonečně častokrát, právě když každá z kopií navštěvuje své akceptující stavy nekonečně častokrát.

Automat rozpoznávající $L_1 \cap L_2$ lze tedy definovat jako $\mathcal{A} = (S, \Sigma, \Delta, S_{in}, F)$, kde

- $S = S_1 \times S_2 \times \{1, 2\}$,
- Δ je definována takto:

$$\begin{array}{lll}
(s_1, s_2, 1) \xrightarrow{a} (s_1', s_2', 1) & \text{je-li} & s_1 \xrightarrow{a} s_1' \in \Delta_1,\ s_2 \xrightarrow{a} s_2' \in \Delta_2,\ s_1 \notin F_1 \\
(s_1, s_2, 1) \xrightarrow{a} (s_1', s_2', 2) & \text{je-li} & s_1 \xrightarrow{a} s_1' \in \Delta_1,\ s_2 \xrightarrow{a} s_2' \in \Delta_2,\ s_1 \in F_1 \\
(s_1, s_2, 2) \xrightarrow{a} (s_1', s_2', 2) & \text{je-li} & s_1 \xrightarrow{a} s_1' \in \Delta_1,\ s_2 \xrightarrow{a} s_2' \in \Delta_2,\ s_2 \notin F_2 \\
(s_1, s_2, 2) \xrightarrow{a} (s_1', s_2', 1) & \text{je-li} & s_1 \xrightarrow{a} s_1' \in \Delta_1,\ s_2 \xrightarrow{a} s_2' \in \Delta_2,\ s_2 \in F_2,
\end{array}$$

- $S_{in} = \{(s_1, s_2, 1) \mid s_1 \in S_{in}^1,\ s_2 \in S_{in}^2\}$,
- $F = S_1 \times F_2 \times \{2\}$.

Uvedený automat akceptuje, jestliže přepíná z druhé do první komponenty nekonečně častokrát. Počet stavů je tedy úměrný součinu počtu stavů automatů $\mathcal{A}_i$. Poznamenejme, že ekvivaletně jsme mohli při konstrukci $\mathcal{A}$ položit $F = F_1 \times S_2 \times \{1\}$).

Snadno se ověří, že $L(\mathcal{A}) = L(\mathcal{A}_1) \cap L(\mathcal{A}_2)$. □

Věta 1.9. *Nechť $U \subseteq \Sigma^*$ je regulární a $L \subseteq \Sigma^\omega$ je Büchi–rozpoznatelný. Pak*

(i) U^ω *je Büchi–rozpoznatelný a*

(ii) $U.L$ *je Büchi–rozpoznatelný.*

Důkaz. (i) Bez újmy na obecnosti můžeme předpokládat, že U neobsahuje prázdné slovo (platí totiž, že $U^\omega = (U - \{\varepsilon\})^\omega$) a že konečný automat $\mathcal{A} = (S, \Sigma, \Delta, q_0, F)$ rozpoznávající U má jediný počáteční stav q_0, do kterého nevedou žádné přechody. Büchiho automat rozpoznávající U^ω je $\mathcal{A}' = (S, \Sigma, \Delta', q_0, \{q_0\})$, kde $\Delta' = \Delta \cup \{s \xrightarrow{a} q_0 \mid s \xrightarrow{a} s', s' \in F\}$.

(ii) Büchiho automat rozpoznávající zřetězení $U.L$ se sestrojí analogicky jako pro případ konečných automatů. Formální popis konstrukce i důkaz její korektnosti je přenechán čtenáři. □

Poznámka 1.10. Uzavřenost vůči doplňku. Büchi–rozpoznatelné jazyky jsou uzavřeny i vůči operaci doplňku, avšak důkaz je obtížnější a bude mu věnována samostatná sekce. Jedním problémem je, že ke konstrukci nemůžeme použít deterministického Büchiho automatu (viz Důsledek 1.7). Druhým problémem je, že samotná Büchiho akceptační podmínka, tj. cyklus v přechodovém grafu automatu při úspěšném běhu na α, může mimo akceptující stavy z F obsahovat též stavy, které nejsou z F, a tudíž nahrazení F jeho komplementem by způsobilo, že α by bylo opět akceptováno. Tento problém bude pro nás (mimo jiné) motivací hledat "jemnější" akceptační podmínky (viz kapitola 2).

Uzávěrové vlastnosti deterministických Büchi–rozpoznatelných jazyků

Řekneme, že jazyk L je deterministický Büchi–rozpoznatelný, jestliže existuje deterministický Büchiho automat $\mathcal{A}$ takový, že $L = L(\mathcal{A})$.

Věta 1.11. *Jsou-li $L_1, L_2 \subseteq \Sigma^\omega$ deterministické Büchi–rozpoznatelné jazyky, pak i jazyky $L_1 \cup L_2$ a $L_1 \cap L_2$ jsou deterministické Büchi–rozpoznatelné.*

Důkaz. Mějme deterministické Büchiho automaty $\mathcal{A}_i = (S_i, \Sigma, \Delta_i, q_i, F_i)$ takové, že $L_i = L(\mathcal{A}_i),\ i = 1, 2$.

Uzavřenost vůči sjednocení: připomeňme (viz věta 1.6), že L je deterministický Büchi–rozpoznatelný jazyk, právě když existuje $U \subseteq \Sigma^*$ taková, že $L = \mathsf{lim}(U)$. Tedy existují $U_i \subseteq \Sigma^*$ taková, že $L_i = \mathsf{lim}(U_i), i = 1, 2$. Tvrzení o *existenci* deterministického automatu okamžitě plyne ze vztahu $\mathsf{lim}(U_1 \cup U_2) = \mathsf{lim}(U_1) \cup \mathsf{lim}(U_2)$. Ke *konstrukci* odpovídajícího automatu $\mathcal{A}$ lze použít synchronního paralelního spojení automatů $\mathcal{A}_i$ s množinou koncových stavů $F = F_1 \times S_2 \cup S_1 \times F_2$. Jelikož $\mathcal{A}_i$ jsou deterministické, je deterministický i $\mathcal{A}$. Poznamenejme, že výsledný $\mathcal{A}$ má počet stavů roven součinu počtů stavů automatů $\mathcal{A}_i$. Dále si všimněme, ke konstrukci $\mathcal{A}$ nelze použít konstrukci uvedenou v důkazu věty. 1.8

Uzavřenost vůči průniku lze dokázat užitím konstrukce z důkazu věty 1.8. Snadno se nahlédne, že pokud jsou dané $\mathcal{A}_i$ deterministické, pak i výsledný automat $\mathcal{A}$ je deterministický. □

1.4 Regulární ω–jazyky

Pro zevrubnější analýzu Büchi–rozpoznatelných jazyků budeme používat toto značení: je-li dán Büchiho automat $\mathcal{A} = (S, \Sigma, \Delta, S_{in}, F)$, pak klademe

$$W_{ss'} = \{w \in \Sigma^* \mid s \xrightarrow{w} s'\} .$$

Zřejmě každý z konečně mnoha jazyků $W_{ss'}$ je regulární (je rozpoznatelný nedeterministickým konečným automatem $(S, \Sigma, \Delta, \{s\}, \{s'\})$). Z definice Büchiho automatu (viz též poznámka 1.2) plyne, že ω–jazyk rozpoznávaný automatem $\mathcal{A}$ je

$$L(\mathcal{A}) = \bigcup_{s_0 \in S_{in}, s \in F} W_{s_0 s}.(W_{ss})^\omega. \tag{1.1}$$

Pokud by $\mathcal{A}$ byl (bez újmy na obecnosti) s jediným počátečním stavem s_0, pak $L(\mathcal{A}) = \bigcup_{s\in F} W_{s_0 s}.(W_{ss})^\omega$.

Věta 1.12. *Libovolný ω–jazyk $L \subseteq \Sigma^\omega$ je Büchi–rozpoznatelný, právě když je konečným sjednocením množin $U.V^\omega$, kde $U, V \subseteq \Sigma^*$ jsou regulární množiny (konečných slov). Navíc lze bez újmy na obecnosti předpokládat, že $V.V \subseteq V$.*

Důkaz. $\Longrightarrow$: platnost této implikace okamžitě plyne ze vztahu (1.1); poznamenejme, že $V.V \subseteq V$ plyne z $W_{ss}.W_{ss} \subseteq W_{ss}$.

$\Longleftarrow$: plyne z uzavřenosti Büchi–rozpoznatelných jazyků vzhledem ke konečnému sjednocení, zřetězení a ω–iteraci. □

Reprezentaci ω–jazyka ve tvaru $L = \bigcup_{i=1}^n U_i V_i^\omega$, kde $U_i, V_i \subseteq \Sigma^*, i = 1, \ldots, n$, jsou regulární výrazy, nazýváme ω–regulárním výrazem.

Řekneme, že jazyk $L \subseteq \Sigma^\omega$ je regulární ω–jazyk (či stručněji je ω–regulární), právě když $L = \bigcup_{i=1}^n U_i V_i^\omega$, kde $U_i, V_i \subseteq \Sigma^*, i = 1, \ldots, n$, jsou regulární jazyky (konečných slov). Viz též následující poznámka 1.13

Zřejmě platí, že jazyk je ω–regulární, právě když je popsán nějakým ω–regulárním výrazem a právě když je Büchi–rozpoznatelný.

Poznámka 1.13. V literatuře se lze setkat s několika definicemi ω–regulárních jazyků, které jsou ekvivaletní s výše uvedenou definicí, která pro naše účely postačí (a je dostatečně jednoduchá). Jedná se například o algebraickou definici, kdy regularitu i ω–regularitu lze definovat pomocí homomorfismu daného jazyka do konečného monoidu. Jiný přístup je založen na definici ω–regulárních jazyků jako nejmenší množiny $\mathcal{R}$ podmnožin Σ^∞ obsahující (i) prázdnou množinu, (ii) jednoprvkové množiny $\{a\}$ pro každý symbol $a \in \Sigma$ a uzavřenou (iii) vůči konečnému sjednocení, zřetězení a (iv) iteracím (*-iterace a ω-iterace) – viz definice zřetězení a ω-iterace v části 1.1.

Konečně poznamenejme, že pro termín regulární ω–jazyk se též používají i termíny racionální či ω–racionální jazyk (či již výše uvedené synonymum ω–regulární jazyk). □

Řekneme, že ω–slovo $\alpha \in \Sigma^\omega$ má nekonečný periodický rozvoj (anglicky “is ultimately periodic”), jestliže existují $u, v \in \Sigma^*$ taková, že $\alpha = uvvv \ldots$.

Věta 1.14. *(i) Libovolný neprázdný ω–regulární jazyk L obsahuje slovo s nekonečným periodickým rozvojem.*

(ii) Problém prázdnosti je pro Büchiho automaty rozhodnutelný.

Důkaz. Tvrzení (i) plyne z rovnosti (1.1) takto: nechť $L = L(A, F)$ pro nějaký Büchiho automat (A, F). Pak $\alpha \in L(A, F)$ je tvaru $\alpha = uv_1v_2 \ldots$, kde $s_0 \stackrel{u}{\to} f$ a $f \stackrel{v_i}{\to} f$ pro všechna $i \geq 1$ a nějaká $s_0 \in S_{in}, f \in F$. Pak i slovo $\beta = uv_1v_1 \ldots \in L(A, F)$ a β je periodické.

Nyní ukážeme, že tvrzení (ii) opět plyne ze vztahu (1.1) a z existence algoritmu pro dosažitelnost akceptujícího stavu v nějaké netriviální (tj. alespoň jednu hranu obsahující) silně souvislé komponentě přechodového grafu Büchiho automatu.

Mějme tedy dán libovolný Büchiho automat $\mathcal{A} = (A, F)$. Pokud v přechodovém systému $A = (S, \Sigma, \Delta, S_{in})$ tohoto automatu abstrahujeme od návěští hran, obdržíme orientovaný graf $G_A = (S, E)$, kde $(s, s') \in E \overset{def}{\iff} \exists a \in \Sigma : s \xrightarrow{a} s'$. Z (1.1) zřejmě plyne, že $L(A, F) \neq \emptyset$, právě když v G_A existuje netriviální silně souvislá komponenta X taková, že X obsahuje nějaký uzel $f \in F$ a X je dosažitelná z nějakého počátečního stavu $s_0 \in S_{in}$. □

Všimněme si, že (při značení jako ve výše uvedeném důkazu) k ověření, zda $L(A, F) \neq \emptyset$, stačí (i) analyzovat *maximální* silně souvislé komponenty orientovaného grafu $G_A = (S, E)$, což lze provést v lineárním čase vzhledem k velikosti grafu (tj. $card(S) + card(E)$) a (ii) ověřit dosažitelnost, což lze opět provést v lineárním čase. Označíme-li $n = card(S)$, pak problém $L(A, F) \neq \emptyset$? je rozhodnutelný, a to v čase $\mathcal{O}(n^2)$.

Poznámka 1.15. Bez důkazu uveďme, že problém neprázdnosti pro Büchiho automaty ($L(A, F) \neq \emptyset$?) je logspace úplný pro NLOGSPACE a že problém neuniversality pro Büchiho automaty ($L(A, F) \neq \Sigma^\omega$?) je logspace úplný pro PSPACE.

I když uzavřenost Büchi–rozpoznatelných jazyků vůči operaci doplňku dokážeme až v následující části 1.5, v zájmu úplnosti námi uváděných výsledků o rozhodnutelných problémech zde zmiňme ještě tyto výsledky:

Důsledek 1.16. *Problém inkluse a ekvivalence je pro Büchiho automaty rozhodnutelný.*

Důkaz. Z uzavřenosti na průnik a komplement (doplněk, zde pro jazyk L značený jako $\neg L$) okamžitě plyne, že oba tyto problémy lze redukovat na problém prázdnosti, protože $L_1 \subseteq L_2 \iff L_1 \cap \neg L_2 = \emptyset$. □

1.5 Komplementace Büchiho automatů a relace kongruence

Jak již bylo naznačeno v poznámce 1.10 je důkaz uzavřenosti Büchi–rozpoznatelných jazyků na komplement netriviální. V literatuře se lze setkat s několika přístupy – zde se budeme držet (do jisté míry) původního Büchiho důkazu; některé ostatní techniky budou naznačeny v dalším textu.

Nejprve nastiňme základní myšlenky Büchiho důkazu: pro daný automat $\mathcal{A} = (S, \Sigma, \Delta, S_{in}, F)$ definujeme kongruenci $\sim_{\mathcal{A}}$ nad Σ^* tak, že dvě (konečná) slova u, v prohlásíme za ekvivalentní, jestliže pro každé $p, q \in S$ platí $p \xrightarrow{u} q \iff p \xrightarrow{v} q$ a navíc při uvedené posloupnosti přechodů pro u lze projít stavem z F, když a jen když je toto možné i pro uvedenou posloupnost přechodů při čtení slova v. Snadno se nahlédne, že $\sim_{\mathcal{A}}$ je kongruence a má pouze konečně mnoho tříd rozkladu (konečný index). Označíme-li symbolem $[u]$ třídu rozkladu obsahující slovo u, ukáže se, že libovolný ω–jazyk tvaru $[u].[v]^\omega$ je buď celý obsažen v $L(\mathcal{A})$, nebo je s ním disjunktní.

Dále lze ukázat[1], že každé ω–slovo lze zapsat jako posloupnost $u_0 u_1 \dots$ konečných slov takových, že všechna u_i, $i \geq 1$, patří do stejné $\sim_{\mathcal{A}}$–třídy. Aplikací tohoto výsledku na ω–slova nepatřící do $L(\mathcal{A})$ se nahlédne, že $\Sigma^\omega - L(\mathcal{A})$ je reprezentovatelná jako sjednocení množin $[u].[v]^\omega$, kde u, v jsou taková, že $u.v^\omega \notin L(\mathcal{A})$. Navíc se jedná o sjednocení

1. původní Büchiho důkaz aplikací Ramsey-ovy věty pro spočetné množiny, zde pomocí lemmatu 1.17

konečně mnoha takových množin (rozklad podle $\sim_A$ má pouze konečně mnoho tříd) a lze jej snadno reprezentovat jako Büchiho automat.

Lemma 1.17. *Nechť $\sim$ je libovolná kongruence na Σ^* s konečným indexem. Pak pro libovolné ω–slovo $\alpha \in \Sigma^\omega$ existují $\sim$–třídy U, V takové, že $\alpha \in U.V^\omega$ (navíc lze předpokládat, že $V.V \subseteq V$).*

Důkaz. Nechť $\sim$ je libovolná kongruence na Σ^* konečného indexu. Pro dané $\alpha \in \Sigma^\omega$ řekneme, že jeho dvě pozice k, k' se *spojují na pozici* m, pro $m > \max\{k, k'\}$, jestliže platí $\alpha[k, m) \sim \alpha[k', m)$. V takovém případě píšeme $k \cong_\alpha^m k'$, či stručněji $k \cong_\alpha k'$, jestliže platí $k \cong_\alpha^m k'$ pro nějaké m.

Poznamenejme, že platí-li $k \cong_\alpha^m k'$, pak pro každé $m' > m$ platí i $k \cong_\alpha^{m'} k'$, protože $\alpha[k, m) \sim \alpha[k', m)$ implikuje $\alpha[k, m)\alpha[m, m') \sim \alpha[k', m)\alpha[m, m')$. Dále si všimněme, že $\cong_\alpha$ je relací ekvivalence s konečným indexem, protože je definována pomocí kongruence $\sim$, která má konečný index. Pak tedy existuje nekonečná rostoucí posloupnost pozic $k_0, k_1, \ldots$, které všechny patří do stejné $\cong_\alpha$–třídy. Můžeme předpokládat, že $k_0 > 0$ a že pro všechna $i > 0$ úseky $\alpha[k_0, k_i)$ patří do stejné $\sim$–třídy, označme ji V (v opačném případě uvážíme vybranou nekonečnou podposloupnost, aby tyto předpoklady byly splněny). Konečně označme U tu $\sim$–třídu, která obsahuje $\alpha[0, k_0)$. Při tomto značení pak platí:

$$\begin{aligned} &\exists k_0.(\ k_0 > 0 \ \wedge \ \alpha[0, k_0) \in U \ \wedge \\ &\wedge \ \forall i > 0.(\ \exists k_i.(\ k_i > k_{i-1} \ \wedge \ \alpha[k_0, k_i) \in V \ \wedge \ k_0 \cong_\alpha k_i\ )))\ . \end{aligned} \tag{1.2}$$

Ukážeme, že $\alpha \in U.V^\omega$. Předpokládejme tedy, že máme k_0 a nekonečnou posloupnost $k_1, k_2, \ldots$ splňující (1.2). Přechodem k vhodné podposloupnosti lze dále předpokládat, že pro všechna $i \geq 0$ se pozice k_0, k_i spojují na nějaké pozici m takové, že $m < k_{i+1}$, a tedy se spojují i na pozici k_{i+1}. Nyní ukážeme, že $\alpha[k_i, k_{i+1}) \in V$ pro všechna $i \geq 0$. Z (1.2) okamžitě máme $\alpha[k_0, k_1) \in V$. Pro $i > 0$ z (1.2) máme $\alpha[k_0, k_{i+1}) \in V$ a víme, že pozice k_0, k_i se spojují na pozici k_{i+1}. Odtud plyne, že $\alpha[k_i, k_{i+1}) \in V$, a tedy $\alpha \in U.V^\omega$.

Nyní zbývá ukázat tvrzení $V.V \subseteq V$. Jelikož V je třída rozkladu podle kongruence, pak stačí ukázat, že $V.V \cap V \neq \emptyset$. Toto však platí, protože úseky $\alpha[k_0, k_i), \alpha[k_i, k_{i+1})$ a $\alpha[k_0, k_{i+1})$ patří do V pro libovolné $i > 0$. □

Saturace. Mějme libovolnou kongruenci $\sim$ na Σ^*. Řekneme, že $\sim$ saturuje ω–jazyk $L \subseteq \Sigma^*$, jestliže pro všechny $\sim$–třídy U, V platí:

$$\forall U, V \in \Sigma^*/_\sim :\ U.V^\omega \cap L \neq \emptyset \implies U.V^\omega \subseteq L\ .$$

Značení. Pro daný Büchiho automat $\mathcal{A} = (S, \Sigma, \Delta, S_{in}, F)$ píšeme $s \xrightarrow[F]{w} s'$, právě když existuje běh automatu $\mathcal{A}$ na slově w ze stavu s do stavu s' takový, že alespoň jeden ze stavů tohoto běhu (včetně stavů s a s') patří do F (srv. se značením $s \xrightarrow{w} s'$). Mimo značení

$$W_{ss'} = \{w \in \Sigma^* \mid s \xrightarrow{w} s'\}\ ,$$

zavedené již v úvodu části 1.4, budeme používat ještě toto značení:

$$W_{ss'}^F = \{w \in \Sigma^* \mid s \xrightarrow[F]{w} s'\}\ .$$

Zřejmě i $W^F_{ss'}$ je regulární (dokažte!).

Kongruence $\sim_A$. Pro daný Büchiho automat $\mathcal{A} = (S, \Sigma, \Delta, S_{in}, F)$ definujme relaci ekvivalence $\sim_A$ na Σ^* takto:

$$u \sim_A v \overset{def}{\Longleftrightarrow} \forall s, s' \in S.(\ (s \overset{u}{\to} s' \iff s \overset{v}{\to} s')\ \wedge\ (s \underset{F}{\overset{u}{\to}} s' \iff s \underset{F}{\overset{v}{\to}} s')\ ),$$

což lze ekvivaletně zapsat, kde $[w]$ značí tu třídu rozkladu $\Sigma^*/_{\sim_A}$ obsahující slovo w, jako:

$$\begin{aligned}[w] \quad = \quad & \textstyle\bigcap_{s,s' \in S} \{W_{ss'} \mid w \in W_{ss'}\} \cap \\ & \textstyle\bigcap_{s,s' \in S} \{W^F_{ss'} \mid w \in W^F_{ss'}\} \cap \\ & \textstyle\bigcap_{s,s' \in S} \{\Sigma^* - W_{ss'} \mid w \notin W_{ss'}\} \cap \\ & \textstyle\bigcap_{s,s' \in S} \{\Sigma^* - W^F_{ss'} \mid w \notin W^F_{ss'}\}\ .\end{aligned}$$

Tedy každá $\sim_A$–třída obsahující slovo w je průnikem všech takových regulárních množin $W_{ss'}, W^F_{ss'}, \Sigma^* - W_{ss'}$ a $\Sigma^* - W^F_{ss'}$, které obsahují slovo w. Tedy z regularity každé $W_{ss'}$ a $W^F_{ss'}$ plyne, že každá $\sim_A$–třída $[w]$ je regulární. Díky konečnosti množiny stavů S má $\sim_A$ konečný index a lze ukázat (dokažte!), že $\sim_A$ je kongruence.

Ukažme nyní, že $L(\mathcal{A})$ lze reprezentovat pomocí $\sim_A$–tříd (pro jisté technické zjednodušení a bez újmy na obecnosti uvažujeme automat s jediným počátečním stavem).

Lemma 1.18. *Nechť $\mathcal{A}$ je Büchiho automat nad Σ. Pak $\sim_A$ saturuje ω–jazyk $L(\mathcal{A})$.*

Důkaz. Nechť $\mathcal{A} = (S, \Sigma, \Delta, \{q_0\}, F)$ je Büchiho automat. Mějme ω–slovo $\alpha = uv_1v_2\ldots$ z jazyka $L(\mathcal{A})$ a $\sim_A$–třídy U, V takové, že $u \in U$ a $v_i \in V, i \geq 1$. Ukážeme, že pak $U.V^\omega \subseteq L(\mathcal{A})$.

Uvažme akceptující běh automatu $\mathcal{A}$ na slově α. V tomto běhu tedy existují stavy $s_1, s_2, \ldots$ takové, že

$$q_0 \overset{u}{\to} s_1 \overset{v_1}{\to} s_2 \overset{v_2}{\to} s_3 \overset{v_3}{\to} \cdots\ ,$$

kde navíc platí

$$s_i \underset{F}{\overset{v_i}{\to}} s_{i+1} \quad \text{pro nekonečně mnoho } i \geq 1\ .$$

Mějme nyní β libovolné takové, že $\beta \in U.V^\omega$. Zřejmě stačí ukázat, že $\beta \in L(\mathcal{A})$. Slovo β lze psát ve tvaru $\beta = u'v'_1v'_2\ldots$, kde $u' \in U, v'_i \in V, i \geq 1$. Dle předpokladu lemmatu jsou U, V $\sim_A$–třídy, a tedy $u \sim_A u', v_i \sim_A v'_i$, takže obdržíme

$$q_0 \overset{u'}{\to} s_1 \overset{v'_1}{\to} s_2 \overset{v'_2}{\to} s_3 \overset{v'_3}{\to} \cdots\ ,$$

kde navíc platí

$$s_i \underset{F}{\overset{v'_i}{\to}} s_{i+1} \quad \text{pro nekonečně mnoho } i \geq 1\ .$$

Takže jsme získali běh automatu $\mathcal{A}$ na slově β, kde se nějaký stav z F vyskytuje nekonečně častokrát, a tedy $\beta \in L(\mathcal{A})$. □

Věta 1.19. *Nechť $L \subseteq \Sigma^\omega$ je Büchi–rozpoznatelný jazyk, pak $\Sigma^\omega - L$ je též Büchi–rozpoznatelný. Je-li dán Büchiho automat rozpoznávající L, pak lze zkonstruovat Büchiho automat rozpoznávající $\Sigma^\omega - L$.*

Důkaz. Spojením Lemmat 1.17 a 1.18 získáváme

$$L = \bigcup_{U,V \in \Sigma^*/\sim_A} \{U.V^\omega \mid U.V^\omega \cap L \neq \emptyset\} . \tag{1.3}$$

Jelikož dle předpokladu má $\sim_A$ konečný index a všechny $\sim_A$–třídy jsou regulární, je L ω–regulární jazyk.

Díky Lemmatu 1.17 a vztahu (1.3) dále získáváme

$$\Sigma^\omega - L = \bigcup_{U,V \in \Sigma^*/\sim_A} \{U.V^\omega \mid U.V^\omega \cap L = \emptyset\} ,$$

a tedy $\Sigma^\omega - L$ je také ω–regulární a tudíž Büchi–rozpoznatelný.

Poznamenejme, že prázdnost průniku $U.V^\omega \cap L$ je rozhodnutelná (viz Věta 1.9 – uzavřenost na zřetězení a ω–iteraci, Věta 1.8 – uzavřenost na průnik a Věta 1.14 – rozhodnutelnost prázdnosti). Užitím Věty 1.12 můžeme tudíž zkonstruovat Büchiho automat $\bar{\mathcal{A}}$ takový, že $L(\bar{\mathcal{A}}) = \Sigma^\omega - L$. □

Výše uvedené Lemma 1.18 lze použít též k formulaci ω–jazyků pomocí konečných pologrup. Zopakujme, že jazyk $W \subseteq \Sigma^*$ je regulární, právě když existuje konečný monoid (pologrupa s jedničkou) M a monoidový homomorfismus $f : \Sigma^* \to M$ takový, že W je sjednocením (jistých) množin $f^{-1}(m)$, kde $m \in M$. Jelikož $\Sigma^*/_{\cong_\alpha}$ je pro libovolný Büchiho automat $\mathcal{A}$ konečným monoidem, dostaneme s použitím Věty 1.12 a Lemmatu 1.18 toto tvrzení:

Věta 1.20. *Jazyk $L \subseteq \Sigma^\omega$ je ω–regulární, právě když existuje konečný monoid a monoidový homomorfismus $f : \Sigma^* \to M$ takový, že L je sjednocením jistých množin $f^{-1}(m).(f^{-1}(e))^\omega$, kde $m, e \in M$ (a kde lze předpokládat, že $e.e = e$).*

Pro případ jazyků konečných slov rovněž připomeňme tzv. prefixovou ekvivalenci $\sim_L$, která je definována takto: Nechť $L \subseteq \Sigma^*$ je libovolný (ne nutně regulární) jazyk. Na Σ^* definujeme relaci $\sim_L$ zvanou prefixová ekvivalence pro L takto: pro $u, v \in \Sigma^*$ klademe

$$u \sim_L v \overset{def}{\iff} \forall w \in \Sigma^* : uw \in L \iff vw \in L.$$

Lze ukázat, že $\sim_L$ je pravou kongruencí, a to největší takovou, která saturuje L (tj. L lze vyjádřit jako sjednocení některých tříd rozkladu určeného jistou pravou kongruencí na Σ^*, přičemž těchto tříd však obecně nemusí být konečně mnoho). V případě regulárních jazyků pak platí (viz Nerodova–Myhillova věta), že L je regulární, právě když $\sim_L$ má konečný index (přičemž $\Sigma^*/_{\sim_L}$, zvaná též syntaktický monoid, odpovídala minimálnímu automatu pro L – počet stavů libovolného minimálního automatu rozpoznávajícího jazyk L je roven indexu prefixové ekvivalence $\sim_L$).

Pro jazyky nekonečných slov lze analogicky definovat na Σ^* prefixovou ekvivalenci $\sim_L$ takto: pro $u, v \in \Sigma^*$ klademe

$$u \sim_L v \overset{def}{\iff} \forall \alpha \in \Sigma^\omega : u\alpha \in L \iff v\alpha \in L ,$$

která je opět pravou kongruencí a dále $\simeq_L$

$$u \simeq_L v \overset{def}{\iff} \forall x, y, z \in \Sigma^* : ((xuyz^\omega \in L \iff xvyz^\omega \in L) \wedge$$
$$(x(yuz)^\omega \in L \iff x(yvz)^\omega \in L)) \ ,$$

o které lze ukázat, že je kongruencí na Σ^* (slova u, v jsou ekvivaletní, právě když je pomocí jazyka L nelze rozlišit jako odpovídající si úseky ω–slov s nekonečným periodickým rozvojem).

Celkem snadno se nahlédne, že ω–regularita L implikuje, že $\sim_L$ i $\simeq_L$ mají konečný index (analogicky jako pro konečná slova: pro libovolný Büchiho automat $\mathcal{A}$ rozpoznávající L má $\sim_A$ konečný index a je zjemněním ekvivalencí $\sim_L$ i $\simeq_L$).

Všimněme si však, že obrácená implikace obecně neplatí:

Poznámka 1.21. Existují neregulární ω–jazyky takové, že $\sim_L$ i $\simeq_L$ mají konečný index.

Důkaz. Pro libovolné dané $\beta \in \Sigma^\omega$ nechť $L(\beta)$ je jazyk všech ω–slov, která mají stejnou příponu jako β. Pak dvě slova u, v jsou $\sim_{L(\beta)}$ ekvivaletní, protože příslušnost ω–slov $u\alpha, v\alpha$ do jazyka $L(\beta)$ vůbec na u, v nezávisí, a tedy $\sim_{L(\beta)}$ má pouze jednu třídu rozkladu.

Zvolíme-li β takové, že nemá nekonečný periodický rozvoj, pak zřejmě $L(\beta)$ není regulární (viz věta 1.14). Dále lze ukázat, že v tomto případě i kongruence $\simeq_{L(\beta)}$ má jen jednu třídu rozkladu. □

Konečně bez důkazu jen zmiňme, že platí "maximalita" kongruence $\simeq_L$, to jest, že jazyk L je ω–regulární, právě když $\simeq_L$ má konečný index a saturuje L; navíc $\simeq_L$ je největší kongruencí saturující L. Tento výsledek nás opravňuje nazvat $\Sigma^*/_{\simeq_L}$ syntaktickým monoidem, kde součinem je zřetězení tříd uvedeného rozkladu.

1.6 Modifikace Büchiho akceptační podmínky

V některých aplikacích, které jsou však mimo rozsah tohoto textu (například vztah Büchiho automatů a jistých temporálních logik), je výhodnější pracovat s jistými modifikacemi Büchiho akceptační podmínky. Uveďme alespoň jednu z nich.

Zobecněný Büchiho automat je ω–automat $\mathcal{A} = (A, \mathcal{G})$, kde $A = (S, \rightarrow, S_{in})$ je LTS nad Σ a $\mathcal{G} = \{G_1, \ldots, G_k \mid G_i \subseteq S\}$. Vstupní ω–slovo α je akceptováno, právě když existuje běh ρ na α takový, že platí: $\forall i. 1 \leq i \leq k : \mathsf{inf}(\rho) \cap G_i \neq \emptyset$.

Büchiho automat je tedy speciálním případem zobecněného Büchiho automatu. Snadno se nahlédne, že díky uzavřenosti (deterministických i nedeterministických) Büchiho automatů vzhledem k operaci průniku se třída akceptovaných jazyků nezmění. Při značení jako výše totiž platí:

$$L(A, \mathcal{G}) \ = \ \bigcap_{i=1}^{k} L(A, G_i) \ .$$

Büchiho automat $\mathcal{A}'$ simulující automat $\mathcal{A} = (A, \mathcal{G})$ lze zkonstruovat takto: položíme $\mathcal{A}' = (S', \Sigma, \rightarrow', {S_{in}}', F)$, kde

$S' = S \times \{1, \ldots, k\}$,
${S_{in}}' = S_{in} \times \{1\}$ a

$F = G_1 \times \{1\}$ a $\to'$ je definována takto:
$(s,j) \stackrel{a}{\to}{}' (t,j)$ je-li $s \stackrel{a}{\to} t,\ s \notin G_j$
$(s,j) \stackrel{a}{\to}{}' (t,i)$ je-li $s \stackrel{a}{\to} t,\ s \in G_j$ a kde $i = (j \bmod k) + 1$.

Ve druhé komponentě stavů je uchováván index $j \in [1,k]$, dokud se neprojde nějakým stavem z G_j. V tom případě je pak j zvýšeno o 1 (modulo k).

Aby některý ze stavů z F byl navštěvován nekonečně často, musí být nějaký stav z každé množiny G_i navštěvován nekonečně často. Obráceně, je-li možné navštívit každou z G_i nekonečně často, je možné to provádět v pořadí $G_1, G_2, \ldots, G_k$, a tedy projít některým stavem z F nekonečně často.

Kapitola 2

Silnější akceptační podmínky

Jak bylo ukázáno v předchozí kapitole, deterministické Büchiho automaty nerozpoznávají celou třídu ω–regulárních jazyků (důsledek 1.7). Pro konstrukci komplementárního automatu tedy nešlo použít deterministický automat. Při použití nedeterministických Büchiho automatů by záměna akceptujích a neakceptujích stavů ke komplementárnímu automatu nevedla – viz též poznámka 1.10. Zavedeme tedy silnější akceptační podmínky, které, jak uvidíme, umožní definovat deterministické automaty rozponávající celou třídu ω–regulárních jazyků.

2.1 Mullerovy automaty

Jednou z možností, jak obejít výše naznačené problémy, je specifikovat akceptační podmínku tak, aby "akceptační cyklus" v automatu obsahoval pouze samé akceptující stavy; je zřejmé, že takto musíme specifikovat všechny akceptační cykly. Následující definici zavedl Muller.

Mullerův automat. Nechť $A = (S, \Delta, S_{in})$ je konečně stavový LTS nad Σ. Řekneme, že ω–automat $\mathcal{A} = (A, Acc)$ je Mullerův automat, právě když Acc je tzv. Mullerova akceptační podmínka (MAC):

je dána množina $\mathcal{F} \subseteq 2^S$ podmnožin množiny stavů S, $\mathcal{F} = \{F_1, F_2, \ldots, F_k \mid F_i \subseteq S\}$, zvaná Mullerova akceptační tabulka.

Slovo $\alpha : \mathbb{N}_0 \to \Sigma$ je akceptováno, právě když existuje běh $\rho : \mathbb{N}_0 \to S$ na slově α takový, že $\mathsf{inf}(\rho) \in \mathcal{F}$, tj. $\mathsf{inf}(\rho) = F_i$ pro nějaké $i \in \{1, 2, \ldots, k\}$.

Tuto podmínku ($\mathcal{A}$ akceptuje α) můžeme formálně zapsat takto:

(MAC) $$\exists \rho.(\rho(0) \in S_{in} \ \wedge \ \forall i.(\rho(i) \stackrel{\alpha(i)}{\to} \rho(i+1) \in \Delta) \ \wedge \ \mathsf{inf}(\rho) \in \mathcal{F})$$

Uvedený Mullerův automat $\mathcal{A}$ značíme $(A, \mathcal{F})$ nebo též $(S, \Sigma, \Delta, S_{in}, \mathcal{F})$. Jazyk rozpoznávaný (též akceptovaný) Mullerovým automatem $\mathcal{A} = (A, \mathcal{F})$ je množina

$$L(\mathcal{A}) = \{\alpha \in \Sigma^\omega \mid \mathcal{A} \text{ akceptuje } \alpha\}$$

všech ω–slov akceptovaných tímto automatem a značíme jej též $L(A, \mathcal{F})$.

Mullerova akceptační podmínka vskutku klade přísnější požavky na úspěšný běh, než podmínka Büchiho: každá množina (položka Mullerovy tabulky) $F \in \mathcal{F}$ klade positivní požadavek na stavy z F (všechny musí být navštěvovány nekonečně mnohokrát) a současně negativní požadavek na stavy z $S - F$ (tyto musí být navštíveny jen konečně mnohokrát). Jinak řečeno, každý úspěšný běh musí od jisté pozice procházet jen přes stavy z F, a to bez přechodu do jakéhokoliv stavu mimo F. Formálně lze podmínku $\mathsf{inf}(\rho) \in \mathcal{F}$ z MAC zapsat jako formuli 1. řádu takto:

$$\bigvee_{F \in \mathcal{F}} \Big(\bigwedge_{q \in F} \forall i \, \exists j.\, j > i : \rho(j) = q \;\wedge \bigwedge_{q \in S - F} \neg \forall i \, \exists j.\, j > i : \rho(j) = q \Big)$$

Příklad 2.1. Připomeňme Büchiho automat $\mathcal{A}_1$ z příkladu 1.4, který akceptoval jazyk $L \subseteq \{a,b\}^\omega$ všech slov takových, která obsahovala nekonečně mnoho výskytů symbolu a. $\mathcal{A}_1 = (\{s_1, s_2\}, \{a,b\}, \Delta_1, \{s_1\}, \{s_1\})$ je *deterministický* automat s *totální* přechodovou funkcí a s Büchiho akceptační podmínkou $\{s_1\}$.

Navrhnout Mullerův automat $\mathcal{A}$ akceptující tentýž L je snadné: daný LTS se vůbec nezmění (tj. zůstává deterministický) a MAC je $\mathcal{F} = \{\{s_1\}, \{s_1, s_2\}\}$. Dále jsme ukázali, že doplněk jazyka L není aceptován žádným deterministickým Büchiho automatem. Snadno se ověří, že doplněk jazyka L akceptuje týž Mullerův automat (tedy s týmž *deterministickým* LTS), který akceptoval L s tím, že akceptační podmínka se změní na $\{\{s_2\}\}$. Je tedy vidět, že deterministické Mullerovy automaty akceptují ostře větší třídu jazyků, než deterministické Büchiho automaty. □

Simulace

Lemma 2.2. *Ke každému Büchiho automatu existuje ekvivaletní Mullerův automat.*

Důkaz. V tvrzení lemmatu obsažená simulace Büchiho automatu automatem Mullerovým je přímočará: v Mullerově akceptační tabulce zřídíme položku pro každou podmnožinu stavů, která obsahuje akceptující stav daného, simulovaného Büchiho automatu.

Nechť (A, G), kde $A = (S, \rightarrow, S_{in})$ je Büchiho automat nad Σ. Simulující Mullerův automat je $(A, \mathcal{F}_G)$, kde $\mathcal{F}_G = \{F \subseteq S \mid F \cap G \neq \emptyset\}$. Snadno se nahlédne, že $L(A, G) = L(A, \mathcal{F}_G)$: libovolný úspěšný běh Büchiho automatu bude vyhovovat jedné z položek v Mullerově akceptační tabulce $\mathcal{F}_G$. Obráceně, jakýkoli běh vyhovující některé položce z $\mathcal{F}_G$ musí projít některým stavem z G nekonečně mnohokrát. □

Z výše uvedené konstrukce je vidět, že simulující $(A, \mathcal{F}_G)$ je deterministický, právě když simulovaný (A, G) je deterministický (výchozí LTS A se nezměnil – tato simulace tedy ani nezavádí, ani neodstraňuje nedeterminismus).

Lemma 2.3. *Ke každému Mullerovu automatu existuje ekvivaletní Büchiho automat.*

Důkaz. Ukážeme, že libovolný Mullerův automat lze simulovat *nedeterministickým* Büchiho automatem. Nechť $(A, \mathcal{F})$ je Mullerův automat, kde $\mathcal{F} = \{F_1, F_2, \ldots, F_k\}$. Nejprve pro každé $i, i = 1, \ldots, k$, zkonstruujeme Büchiho automat $\mathcal{A}_i = (A_i, G_i)$ takový, že $\mathcal{A}_i$ akceptuje vstup α, právě když existuje běh ρ automatu $(A, \mathcal{F})$ na α s vlastností $\mathsf{inf}(\rho) = F_i$.

Při simulaci běhu automatu $(A, \mathcal{F})$ na slově α simulující $\mathcal{A}_i$ nedeterministicky uhodne, že již nebudou navštíveny žádné stavy z $S - F_i$; následně tedy musí kontrolovat korektnost své volby: simuluje (či přesněji: je schopen simulovat) pouze ty přechody, které "nevybočí" mimo F_i. Současně $\mathcal{A}_i$ musí při cyklení přes množinu stavů F_i kontrolovat, zda jsou vskutku *všechny* stavy z F_i navštěvovány nekonečně častokrát.

Nechť $A = (S, \rightarrow, S_{in})$ a $F_i = \{s_{i_1}, s_{i_2}, \ldots, s_{i_m}\}$. Konstruujeme $\mathcal{A}_i = (A_i, G_i)$, kde $A_i = (S_i, \rightarrow_i, S^i_{in})$ a G_i jsou definovány takto:

- $S_i = S \cup \{(s, \mathsf{cyklus}_i, j) \mid s \in F_i, j \in \{0, 1, \ldots, m-1\}\}$,
- relace přechodu $\rightarrow_i$ je definována takto (upozorněme, že ve třetím a čtvrtém řádku definice přechodové relace $\rightarrow_i$ je index $j+1$ u stavu $s_{i_{j+1}}$ počítán modulo m):

$$
\begin{array}{ll}
s \xrightarrow{a}_i s' & \text{je-li} \quad s \xrightarrow{a} s' \\
s \xrightarrow{a}_i (s', \mathsf{cyklus}_i, 0) & \text{je-li} \quad s \xrightarrow{a} s',\ s' \in F_i \\
(s, \mathsf{cyklus}_i, j) \xrightarrow{a}_i (s', \mathsf{cyklus}_i, j) & \text{je-li} \quad s \xrightarrow{a} s',\ s' \in F_i,\ s' \neq s_{i_{j+1}} \\
(s, \mathsf{cyklus}_i, j) \xrightarrow{a}_i (s', \mathsf{cyklus}_i, (j+1) \bmod m) & \text{je-li} \quad s \xrightarrow{a} s',\ s' \in F_i,\ s' = s_{i_{j+1}}
\end{array}
$$

- $S^i_{in} = S_{in}$
- $G_i = \{(s_{i_m}, \mathsf{cyklus}_i, m-1)\}$

Hledaný Büchiho automat simulující daný automat Mullerův je zřejmě sjednocením všech právě zkonstruovaných $\mathcal{A}_i = (A_i, G_i), i = 1, \ldots, k$ – jeho existence i konstrukce je dána větou 1.8 o uzavřenosti Büchi–rozpoznatelných jazyků vůči sjednocení. Snadno se nahlédne, že $L(A, \mathcal{F}) = \bigcup_{i=1}^{k} L(A_i, G_i)$. $\square$

Z právě uvedených lemmat plyne, že třída ω–jazyků rozpoznatelných Mullerovými automaty je totožná s třídou Büchi–rozpoznatelných (tj. ω–regulárních) jazyků. Příklad 2.1 klade přirozenou otázku, zda *deterministické* Mullerovy automaty rozpoznávají celou třídu ω–regulárních jazyků; velmi netriviální důkaz, že tomu tak je, podal McNaughton. Větu, která nese jeho jméno, uvádíme bez důkazu.

Věta 2.4. (McNaughton). *Každý ω–regulární jazyk je rozpoznatelný nějakým deterministickým Mullerovým automatem.*

Povšimněme si, že McNaughtonova věta spolu s oběma simulacemi uvedenými v lemmatech 2.2 a 2.3 podává (alternativní) konstrukci komplementárního automatu pro daný Büchiho automat. Důvod je v tom, že k danému (nyní bez újmy na obecnosti) *deterministickému* Mullerovu automatu $(A, \mathcal{F})$ s *totální* přechodovou funkcí lze komplementární (doplněk rozpoznávající) automat zkonstruovat velmi snadno: v (MAC) namísto $\mathcal{F}$ stačí položit $\overline{\mathcal{F}} = 2^S - \mathcal{F}$. Získáme tedy Mullerův automat $(A, \overline{\mathcal{F}})$, kde $\overline{\mathcal{F}} = \{F \subseteq S \mid F \notin \mathcal{F}\}$. Snadno se nahlédne, že $L(A, \overline{\mathcal{F}}) = \Sigma^\omega - L(A, \mathcal{F})$.

Klíčovým a nejtěžším místem zůstává převod (obecně) nedeterministického Büchiho automatu na nějaký (ne nutně Mullerův) deterministický ω–automat takový, že deterministické ω–automaty tohoto typu rozpoznávají všechny ω–regulární jazyky.

2.2 Rabinovy a Streettovy automaty

Důkaz McNaughtonovy věty bývá v dnešní době podáván nepřímo pomocí tzv. Safrovy konstrukce, která převádí libovolný nedeterministický Büchiho automat na deterministický ω–automat s tzv. Rabinovou akceptační podmínkou – o těchto automatech pojednává bezprostředně následující sekce.

2.2.1 Rabinův automat

Níže uvedenou akceptační podmínku pomocí tzv. tabulky dvojic (anglicky "pairs table") zavedl poprvé Rabin.

Rabinův automat. Nechť $A = (S, \Delta, S_{in})$ je konečně stavový LTS nad Σ. Řekneme, že ω–automat $\mathcal{A} = (A, Acc)$ je Rabinův automat, právě když Acc je tzv. Rabinova akceptační podmínka (RAC):

je dána množina $\mathcal{PT} \subseteq 2^S \times 2^S$ dvojic podmnožin množiny stavů S (značená $\mathcal{PT} = \{(G_1, R_1), (G_2, R_2), \ldots, (G_k, R_k)\}$ a zvaná též Rabinova tabulka).
Slovo $\alpha : \mathbb{N}_0 \to \Sigma$ je akceptováno, právě když existuje běh $\rho : \mathbb{N}_0 \to S$ na slově α takový, že $\exists i.1 \leq i \leq k : \mathsf{inf}(\rho) \cap G_i \neq \emptyset \ \wedge \ \mathsf{inf}(\rho) \cap R_i = \emptyset$.

Tuto podmínku ($\mathcal{A}$ akceptuje α) můžeme formálně zapsat takto:

$$\text{(RAC)} \qquad \begin{aligned} \exists \rho.(\rho(0) \in S_{in} \wedge \forall i.(\rho(i) \stackrel{\alpha(i)}{\to} \rho(i+1) \in \Delta) \\ \wedge \exists i.\, 1 \leq i \leq k : \mathsf{inf}(\rho) \cap G_i \neq \emptyset \wedge \ \mathsf{inf}(\rho) \cap R_i = \emptyset) \end{aligned}$$

Uvedený Rabinův automat $\mathcal{A}$ značíme $(A, \mathcal{PT})$ nebo též $(S, \Sigma, \Delta, S_{in}, \mathcal{PT})$. Jazyk rozpoznávaný (též akceptovaný) Rabinovým automatem $\mathcal{A} = (A, \mathcal{PT})$ je množina

$$L(\mathcal{A}) = \{\alpha \in \Sigma^\omega \mid \mathcal{A} \text{ akceptuje } \alpha\}$$

všech ω–slov akceptovaných tímto automatem a značíme jej též $L(A, \mathcal{PT})$.

Každá dvojice (G_i, R_i) v tabulce dvojic Rabinova automatu tedy specifikuje (podobně jako v Mullerově automatu) positivní a negativní požadavky kladené na úspěšný běh. Positivní podmínka kladená na stavy z G_i je totožná s Büchiho podmínkou. Negativní podmínka kladená na stavy z R_i odpovídá podmínce pro konečně mnoho výskytů stavů z $S - F_i$, uvažujeme-li v Mullerově akceptační tabulce položku F_i. (Pokud si představíme G_i a R_i jako zelená resp. červená světla typu i, pak běh ρ splňuje (G_i, R_i), jestliže některé zelené světlo typu i (tj. z G_i) bliká nekonečně častokrát a současně červené světlo typu i (tj. z R_i) zablikalo jen konečně mnohokrát.)

Podmínku $\exists r.\, 1 \leq r \leq k : \mathsf{inf}(\rho) \cap G_r \neq \emptyset \wedge \ \mathsf{inf}(\rho) \cap R_r = \emptyset$ z RAC lze zapsat jako formuli 1. řádu takto:

$$\bigvee_{r \in \{1,\ldots,k\}} \Big(\bigvee_{q \in G_r} \forall i\, \exists j.\, j > i : \rho(j) = q \ \wedge \ \bigwedge_{q \in R_r} \neg \forall i\, \exists j.\, j > i : \rho(j) = q \Big)$$

Příklad 2.5. Vrátíme-li se opět k příkladu 1.4, pak $L(\mathcal{A}_1)$ (nekonečně mnoho výskytů 'a') je akceptován Rabinovým automatem, který má LTS jako $\mathcal{A}_1$, s tabulkou obsahující jedinou dvojici $(\{s_1\}, \emptyset)$. Doplněk $L(\mathcal{A}_1)$ je akceptován Rabinovým automatem, který má opět tentýž LTS a jeho tabulka obsahuje jedinou dvojici $(\{s_2\}, \{s_1\})$.

Simulace

Büchiho automaty lze jednoduše simulovat pomocí Rabinových automatů: je-li (A, F) Büchiho automat, pak ekvivaletní Rabinův automat je $(A, \mathcal{PT}_F)$, kde $\mathcal{PT}_F = \{\{F, \emptyset\}\}$. Tedy Büchiho automat je speciálním případem Rabinova automatu – stačí totiž položit $k = 1,\ G_1 = F,\ R_1 = \emptyset$.

V obráceném směru můžeme každý Rabinův automat simulovat Büchiho automatem pomocí konstrukce, která je podobná té, jež byla užita při simulaci Mullerova automatu Büchiho automatem (viz lemma 2.3). Opět stačí, díky uzavřenosti Büchiho automatů vůči sjednocení, zkonstruovat pro každou položku $(R_i, G_i) \in \mathcal{PT}$ jeden ("izolovaný") Büchiho automat (A_i, F_i). Každý (A_i, F_i) opět nedeterministicky uhodne, kdy se již žádný ze stavů z R_i neobjeví a následně toto kontroluje spolu s tím, že alespoň jeden ze stavů z G_i se objevuje v běhu nekonečně mnohokrát. Nastíněnou konstrukci nyní popišme formálně.

Nechť $A = (S, \rightarrow, S_{in})$ a $\mathcal{PT} = \{(G_1, R_1), (G_2, R_2), \ldots, (G_k, R_k)\}$. Konstruujeme $\mathcal{A}_i = (A_i, F_i)$, kde $A_i = (S_i, \rightarrow_i, S^i_{in})$ a G_i jsou definovány takto:

- $S_i = S \cup \{(s, \mathsf{cyklus}_i, j) \mid s \in S - R_i,\ j \in \{0, 1\}\ \}$,
- relace přechodu $\rightarrow_i$ je definována takto:

$$
\begin{array}{ll}
s \xrightarrow{a}_i s' & \text{je-li}\ \ s \xrightarrow{a} s' \\
s \xrightarrow{a}_i (s', \mathsf{cyklus}_i, 0) & \text{je-li}\ \ s \xrightarrow{a} s',\ s' \notin R_i \\
(s, \mathsf{cyklus}_i, 0) \xrightarrow{a}_i (s', \mathsf{cyklus}_i, 0) & \text{je-li}\ \ s \xrightarrow{a} s',\ s' \notin R_i,\ s' \notin G_i \\
(s, \mathsf{cyklus}_i, 0) \xrightarrow{a}_i (s', \mathsf{cyklus}_i, 1) & \text{je-li}\ \ s \xrightarrow{a} s',\ s' \notin R_i,\ s' \in G_i \\
(s, \mathsf{cyklus}_i, 1) \xrightarrow{a}_i (s', \mathsf{cyklus}_i, 0) & \text{je-li}\ \ s \xrightarrow{a} s',\ s' \notin R_i
\end{array}
$$

- $S^i_{in} = S_{in}$
- $F_i = \{(s, \mathsf{cyklus}_i, 1) \mid s \in S - R_i\}$

Poznámka 2.6. Dále si všimněme, že libovolný Rabinův automat je speciálním případem automatu Mullerova – každá dvojice $(G_i, R_i) \in \mathcal{PT}$ generuje "částečnou" Mullerovu tabulku $\mathcal{F}_i = \{F \subseteq S \mid F \cap G_i \neq \emptyset, F \cap R_i = \emptyset\}$ a hledaná Mullerova tabulka je $\mathcal{F} = \cup_{i \in \{1,\ldots,k\}} \mathcal{F}_i$, což lze zapsat stručněji jako

$$\mathcal{F} = \{F \subseteq S \mid \bigvee_{i=1}^{k} (F \cap G_i \neq \emptyset\ \wedge\ F \cap R_i = \emptyset)\}\ .$$

Jelikož jsme LTS nijak nemodifikovali, platí, že simulující automat je deterministický, právě když simulovaný automat je deterministický.

Konečně poznamenejme, že pro simulaci Mullerova automatu automatem Rabinovým je nutné použít konstrukci, která je analogická konstrukci použité při simulaci Mullerova automatu Büchiho automatem. Tato konstrukce vnáší nedeterminismus. Není známa nějaká přímočará konstrukce, která by byla simulací deterministického Mullerova automatu deterministickým Rabinovým automatem. Nicméně zdůrazněme (opět s odkazem na Safra-ovu konstrukci a Poznámku 2.6), že deterministické Rabinovy automaty rozpoznávají celou třídu ω–regulárních jazyků.

2.2.2 Streettův automat

Níže uvedenou akceptační podmínku, definovanou opět pomocí tabulky dvojic, zavedl poprvé Streett (1988). Při nekonečných výpočtech jsou tyto automaty vhodné pro specifikaci tzv. "fairness" podmínek, jako je například podmínka typu "požaduje-li proces přidělení zdroje nekonečně častokrát, pak systém garantuje splnění tohoto požadavku rovněž nekonečně častokrát".

Streettův automat. Streettův automat je ω–automat definovaný jako dvojice $(A, \mathcal{PT})$, kde $A = (S, \Delta, S_{in})$ a $\mathcal{PT} = \{(G_1, R_1), (G_2, R_2), \ldots, (G_k, R_k)\}$ jsou definovány tak, jako u Rabinova automatu. Streettova akceptační podmínka (SAC) je definována takto:

> Slovo $\alpha : \mathbb{N}_0 \to \Sigma$ je akceptováno, právě když existuje běh $\rho : \mathbb{N}_0 \to S$ na slově α takový, že $\forall i.1 \leq i \leq k : \mathsf{inf}(\rho) \cap G_i \neq \emptyset \implies \mathsf{inf}(\rho) \cap R_i \neq \emptyset$.

Tuto podmínku můžeme formálně zapsat takto:

(SAC)
$$\begin{aligned} &\exists \rho.(\rho(0) \in S_{in} \wedge \forall i.(\rho(i) \stackrel{\alpha(i)}{\to} \rho(i+1) \in \Delta) \\ &\qquad \wedge \forall i.\, 1 \leq i \leq k : \mathsf{inf}(\rho) \cap G_i \neq \emptyset \implies \mathsf{inf}(\rho) \cap R_i \neq \emptyset) \end{aligned}$$

Uvedený Streettův automat $\mathcal{A}$ (opět) značíme $(A, \mathcal{PT})$ nebo též $(S, \Sigma, \Delta, S_{in}, \mathcal{PT})$. Je tedy vždy nutno uvést, zda se jedná o interpretaci $\mathcal{PT}$ jako tabulky Rabinovy, či Streettovy, pokud to není zřejmé z kontextu. Pokud nebude tato interpretace zadána, budeme o $(A, \mathcal{PT})$ mluvit jako o automatu s tabulkou dvojic.

Všimněme si, že definice SAC přímo koresponduje uvedenému typu fairness podmínek: jestliže je některý ze stavů z G_r navštěvován nekonečně mnohokrát, pak musí v R_r existovat nějaký stav, který je navštěvován rovněž nekonečně mnohokrát a splnění této podmínky je požadováno pro všechna $r,\ 1 \leq r \leq k$.

Podmínku $\forall r.\, 1 \leq r \leq k : \mathsf{inf}(\rho) \cap G_r \neq \emptyset \implies \mathsf{inf}(\rho) \cap R_r \neq \emptyset$ ze SAC lze zapsat jako formuli 1. řádu takto:

$$\bigwedge_{r \in \{1,\ldots,k\}} (\neg \bigvee_{q \in G_r} \forall i\, \exists j.\, j > i : \rho(j) = q \ \vee \ \bigvee_{q \in R_r} \forall i\, \exists j.\, j > i : \rho(j) = q)\ .$$

Přímo z definice Streettova automatu se vidí jeho vztah k automatu Rabinovu:

Tvrzení 2.7. *Nechť $\mathcal{A} = (A, \mathcal{PT})$ je automat s tabulkou dvojic. Nechť $L_R = L(A, \mathcal{PT})$, kde $\mathcal{PT}$ je interpretována jako Rabinova akceptační podmínka, a $L_S = L(A, \mathcal{PT})$, kde $\mathcal{PT}$ je interpretována jako Streettova akceptační podmínka. Pak L_S je doplňkem L_R.*

Simulace

Každý Büchiho automat $(S, \Sigma, \to, S_{in}, F)$ je opět specálním případem Streettova automatu – stačí položit $k = 1, G_1 = S, R_1 = F$ bez změny LTS (tj. $PT_F = \{(S, F)\}$ pro Streettovu interpretaci). O konstrukci Büchiho automatu simulujícího libovolný daný Streettův automat pojednává následující lemma (M.Vardi).

Lemma 2.8. *Nechť $\mathcal{A} = (A, \mathcal{PT})$ je Streettův automat nad Σ, kde $A = (S, \to, S_{in})$, $\mathcal{PT} = \{(G_1, R_1), (G_2, R_2), \ldots, (G_k, R_k)\}$ a $n = card(S)$. Pak lze zkonstruovat Büchiho automat $\mathcal{A}' = (S', \Sigma, \to', S'_{in}, F)$ takový, že $L(\mathcal{A}) = L(\mathcal{A})'$ a $card(S') = n \cdot 2^{\mathcal{O}(k)}$.*

Důkaz. Jako obvykle $\mathcal{A}'$, který simuluje automat $\mathcal{A}$, hádá pozici v běhu ρ takovou, že každý stav navštívený za touto pozicí bude de facto navštíven nekonečně častokrát; od této pozice dál musí $\mathcal{A}'$ kontrolovat splnění akceptační podmínky pro každou dvojici $(G_i, R_i) \in \mathcal{PT}$. Automat $\mathcal{A}'$ tedy musí za uvedenou pozicí kontrolovat pro každé i implikaci: jestliže se některý ze stavů z G_i vyskytuje nekonečně častokrát, pak i některý ze stavů z R_i se vyskytuje nekonečně častokrát. Toto lze realizovat tak, že $\mathcal{A}'$ udržuje, jako komponenty svého stavu, dvě množiny: v první z nich udržuje seznam indexů i takových, že některý stav z G_i je navštěvován nekonečně častokrát, kdežto ve druhé z nich seznam indexů navštěvovaných stavů z R_i. Jakmile se první množina stane podmnožinou množiny druhé, je tato druhá množina nastavena na prázdnou množinu. Je vidět, že akceptační podmínka bude splněna, právě když je zmíněná druhá množina nastavena na prázdnou množinu nekonečně častokrát.

Formálně lze naznačenou konstrukci simulujícího automatu $\mathcal{A}' = (S', \Sigma, \rightarrow', S'_{in}, F)$ zapsat takto:

- $S_i = S \cup \{(s, X, Y) \mid s \in S,\ X, Y \subseteq \{1, 2, \ldots, k\}\ \}$,
- relace přechodu $\rightarrow'$ je definována takto:

$$
\begin{array}{lll}
s \overset{a}{\rightarrow}{}' s' & \text{je-li} & s \overset{a}{\rightarrow} s' \ , \\[1ex]
s \overset{a}{\rightarrow}{}' (s', \emptyset, \emptyset) & \text{je-li} & s \overset{a}{\rightarrow} s',\ s' \notin R_i \ , \\[1ex]
(s, X, Y) \overset{a}{\rightarrow}{}' (s', X \cup G, Y \cup R) & \text{je-li} & s \overset{a}{\rightarrow} s',\ X \cup G \not\subseteq Y \cup R, \text{ kde} \\
 & & G = \{i \mid s' \in G_i, 1 \le i \le k\} \text{ a} \\
 & & R = \{i \mid s' \in R_i, 1 \le i \le k\} \ , \\[1ex]
(s, X, Y) \overset{a}{\rightarrow}{}' (s', X \cup G, \emptyset) & \text{je-li} & s \overset{a}{\rightarrow} s',\ X \cup G \subseteq Y \cup R \ ,
\end{array}
$$

- $S'_{in} = S_{in}$,
- $F = \{(s, X, \emptyset) \mid s \in S, X \subseteq \{1, 2, \ldots, k\}\ \}$.

□