22 Regulární výrazy a regulární jazyky, Kleeneova věta. Algoritmická složitost úloh souvisejících s regulárními jazyky. (A4B01JAG)

22.1 Regulární jazyky

Regulární jazyky viz kapitola 21 Deterministický konečný automat, jazyk přijímaný konečným automatem.

22.1.1 Uzávěrkové vlastnosti třídy regulárních jazyků

Třída regulárních jazyků je uzavřena na sjednocení, průnik, doplněk i rozdíl.

Přesněji, jestliže L_1 a L_2 jsou regulární jazyky, pak také $L_1 \cup L_2$, $L_1 \cap L_2$, $\bar{L_1} = \Sigma^\star \setminus L_1$ a $L_1 \setminus L_2$ jsou také regulární jazyky.

22.1.1.1 Zřetězení jazyků

Jsou dány jazyky L_1 a L_2 nad abecedou Σ. *Zřetězení* jazyků L_1 a L_2 je jazyk $L_1 L_2$ definovaný

$$L_1 L_2 = \{uv | u \in L_1, v \in L_2\}.$$

Tvrzení: Třída regulárních jazyků je uzavřena na zřetězení. Přesněji, jsou-li jazyky L_1 a L_2 regulární, je regulární i jazyk $L_1 L_2$.

22.1.1.2 Operace $\star$

Je dán jazyk L nad abecedou Σ. Definujeme $L_0 = \{\epsilon\}$, $L^{i+1} = L^i L$ pro $i \geq 0$. Pak operace $\star$ pro jazyk L ($L^\star$) je definována

$$L^\star = \{\epsilon\} \cup L \cup L^2 \cup L^3 \cup \ldots = \bigcup_{i=0}^{\infty} L^i.$$

Poznamenejme, že operaci $\star$ se též říká *Kleeneho operátor*.

Tvrzení: Třída regulárních jazyků je uzavřena na operaci $\star$. Přesněji, je-li jazyk L regulární, je regulární i jazyk $L^\star$.

22.2 Regulární výrazy

Regulární výrazy slouží k ještě jinému popisu regulárních jazyků. Právě regulární výrazy daly jméno třídě jazyků přijímaných konečnými automaty (ať už deterministickými nebo nedeterministickými).

22.2.0.3 Regulární výrazy nad abecedou

Je dána abeceda Σ. Množina všech regulárních výrazů nad Σ je definována:

- $\emptyset$ je regulární výraz,
- ϵ je regulární výraz,
- $\mathbf{a}$ je regulární výraz pro každé písmeno $a \in \Sigma$,
- jsou-li $\mathbf{r_1}$ a $\mathbf{r_2}$, pak $\mathbf{r_1} + \mathbf{r_2}$, $\mathbf{r_1 r_2}$ a $\mathbf{r_1^\star}$ jsou regulární výrazy.

22.2.0.4 Jazyk odpovídající regulárnímu výrazu

Každému regulárnímu výrazu nad abecedou Σ odpovídá jazyk nad abecedou Σ a to takto:

- Regulárnímu výrazu $\emptyset$ odpovídá jazyk $\emptyset$.
- Regulárnímu výrazu ϵ odpovídá jazyk $\{\epsilon\}$.
- Je-li $a \in \Sigma$, pak regulárnímu výrazu $\mathbf{a}$ odpovídá jazyk $\{a\}$.
- Jestliže regulárnímu výrazu $\mathbf{r_1}$ odpovídá jazyk L_1 a regulárnímu výrazu $\mathbf{r_2}$ odpovídá jazyk L_2, pak regulárnímu výrazu $\mathbf{r_1} + \mathbf{r_2}$ odpovídá jazyk $L_1 \cup L_2$ a regulárnímu výrazu $\mathbf{r_1 r_2}$ odpovídá jazyk $L_1 L_2$.
- Jestliže regulárnímu výrazu $\mathbf{r}$ odpovídá jazyk L, pak regulárnímu výrazu $\mathbf{r}^\star$ odpovídá jazyk $L^\star$.

Věta: Každý regulární výraz nad abecedou Σ odpovídá regulárnímu jazyku (nad abecedou Σ), tj. jazyku, který je přijímán konečným automatem.

Důkaz: Regulárním výrazům $\emptyset$, ϵ, $\mathbf{a}$ (pro $a \in \Sigma$) odpovídají po řadě jazyky $\emptyset$, $\{\epsilon\}$, $\{a\}$. Všechny tyto jazyky jsou přijímány konečným automatem.

O třídě jazyků přijímaných konečnými automaty víme, že je uzavřena na sjednocení, zřetězení a Kleeneho operaci $\star$. To znamená, že jsou-li jazyky odpovídající regulárním výrazům $\mathbf{r}$, $\mathbf{r_1}$ a $\mathbf{r_2}$ přijímány konečnými automaty, pak totéž platí i pro jazyky odpovídající regulárním výrazům $\mathbf{r_1} + \mathbf{r_2}$, $\mathbf{r_1 r_2}$ a $\mathbf{r}^\star$.

22.2.0.5 Kleeneho věta

Každý jazyk přijímaný konečným automatem je možné popsat regulárním výrazem.

Důkaz: Je dán DFA $M = (Q, \Sigma, \delta, q_0, F)$, který přijímá jazyk L. Pro jednoduchost označme množinu stavů $Q = \{1, \dots, n\}$ a počátěčni stav $q_0 = 1$. Pro $k = 0, 1, \dots, n$ definujeme množiny slov $R_{i,j}^{(k)}$ takto

$R_{i,j}^{(k)}$ je množina těch slov w, které $\delta^\star(i, w) = j$ a sled z i do j prochází pouze přes stavy $1, \dots, k$.

Platí $R_{i,j}^{(0)} = \{a \in \Sigma | \delta(i, a) = j\}$, což je konečná množina písmen. Proto umíme množinu $R_{i,j}^{(0)}$ popsat regulárním výrazem.

Jestliže všechny množiny slov $R_{i,j}^{(k)}$ umíme popsat regulárním výrazem $\mathbf{r}_{i,j}^k$, pak pro množinu slov $R_{i,j}^{(k+1)}$ platí

$$R_{i,j}^{(k+1)} = R_{i,j}^{(k)} \cup R_{i,k+1}^{(k)} \left(R_{k+1,k+1}^{(k)}\right)^\star R_{k+1,j}^{(k)}.$$

Tedy $R_{i,j}^{(k+1)}$ popíšeme regulárním výrazem $\mathbf{r}_{i,j}^k + \mathbf{r}_{i,k+1}^k \left(\mathbf{r}_{k+1,k+1}^k\right)^\star \mathbf{r}_{k+1,j}^k$, což je opět regulární výraz.

Navíc jazyk L je sjednocení všech množin $R_{1,j}^{(n)}$ pro $j \in F$. Proto jazyku L odpovídá regulární výraz $\Sigma_{j \in F} \mathbf{r}_{1,j}^n$.

22.2.0.6 Aplikace regulárních výrazů

1. Program *grep* (Global search for Regular Expression and Print).
2. Využití v editorech.
3. Využití v programovacích jazycích.
4. Využití při syntaktické analýze v překladačích.

Poznámka: Zavedli jsme regulární výrazy tak, jak jsou definovány v teorii konečných automatů. Při praktickém použití regulárních výrazů v computer science se používá jiné značení, a navíc se zavádí rozšířené regulární výrazy, které pak už nepopisují jen regulární jazyky. Víc o těchto regulárních výrazech najdete na webové stránce Pavla Satrapy *http://www.nti.tul.cz/ satrapa/docs/regvyr/*.

22.2.0.7 Některé rovnosti mezi regulárními výrazy

Jsou-li $\mathbf{r}$, $\mathbf{p}$ a $\mathbf{q}$ regulární výrazy, pak platí následující rovnosti (to znamená: regulární výraz odpovídající levé straně a regulární výraz odpovídající pravé straně popisují stejný jazyk):

1. $\mathbf{p} + \mathbf{q} = \mathbf{p} + \mathbf{q}$,
2. $\mathbf{r}\ (\mathbf{p} + \mathbf{q}) = \mathbf{r}\ \mathbf{p} + \mathbf{r}\ \mathbf{q}$,

3. $(\mathbf{p}+\mathbf{q})\,\mathbf{r}=\mathbf{p}\;\mathbf{r}+\mathbf{q}\;\mathbf{r}$,
4. $(\mathbf{r}^\star)^\star=\mathbf{r}^\star$,
5. $(\mathbf{p}+\mathbf{q})^\star=(\mathbf{p}^\star\mathbf{q}^\star)^\star$,
6. $(\mathbf{p}+\mathbf{q})^\star=(\mathbf{p}^\star+\mathbf{q})^\star$,
7. $(\mathbf{p}+\mathbf{q})^\star=(\mathbf{p}^\star\mathbf{q})^\star\,\mathbf{p}^\star$,
8. $\mathbf{r}^\star=\epsilon+\mathbf{r}\mathbf{r}^\star$,
9. $\mathbf{r}\mathbf{r}^\star=\mathbf{r}^\star\mathbf{r}$,
10. $(\mathbf{p}\mathbf{q})^\star=\epsilon+\mathbf{p}\,(\mathbf{q}\mathbf{p})^\star\,\mathbf{q}$,
11. $(\mathbf{p}\mathbf{q})^\star\,\mathbf{p}=\mathbf{p}\,(\mathbf{q}\mathbf{p})^\star$.

22.3 Další uzávěrkové vlastnosti třídy regulárních jazyků

22.3.0.8 Homomorfismus

Jsou dány dvě abecedy Σ, Γ a zobrazení h, které každému písmenu $a\in\Sigma$ přiřadí slovo $h(a)$ nad abecedou Γ.

Zobrazení h rozšíříme na zobrazení, které každému slovu $u\in\Sigma^\star$ přiřazuje slovo nad Γ takto:

- $h(\epsilon)=\epsilon$,
- $h(ua)=h(u)\,h(a)$.

Obraz jazyka L nad Σ je definován $h(L)=\cup\{h(w)\,|w\in L\}$.

Takto definované zobrazení h se nazývá *homomorfismus*.

Příklad: $\Sigma=\{0,1\}$, $\Gamma=\{a,b\}$ a $h(0)=ab^2$, $h(1)=bab$. Pak $h(010)=ab^2babab^2=ab^2\,(ba)^2\,b^2$. Homomorfní obraz jazyka $L=\left\{10^k|k\geq 0\right\}$ je $h(L)=\left\{bab\left(ab^2\right)^k|k\geq 0\right\}$.

22.3.0.9 Substituce

Obecnější pojem než homomorfismus je tzv. substituce. Jsou dány dvě abecedy Σ, Γ a zobrazení σ, které každému písmenu $a\in\Sigma$ přiřadí jazyk nad abecedou Γ.

Analogicky jako pro homomorfismus zobrazení σ rozšíříme na zobrazení, které každému slovu $u\in\Sigma^\star$ přiřazuje jazyk nad Γ takto:

- $\sigma(\epsilon)=\{\epsilon\}$,
- $\sigma(ua)=\sigma(u)\,\sigma(a)$.

Obraz jazyka L nad Σ je $\sigma(L)=\cup\{\sigma(w)\,|w\in L\}$.

Takto definované zobrazení σ se nazývá *substituce*.

Příklad: $\Sigma=\{0,1\}$, $\Gamma=\{a,b\}$, $\sigma(0)=L_1=\{a^n|n\geq 0\}$, $\sigma(1)=L_2=\{b^n|n\geq 0\}$. Pak $\sigma(01)=L_1L_2=\{a^nb^m|n,m\geq 0\}$.

22.3.0.10 Věta

Třída regulárních jazyků je uzavřena na homomorfismy. Jinými slovy, jestliže L je regulární jazyk nad abecedou Σ a h je homomorfismus z Σ do Γ, pak $h(L)$ je regulární jazyk nad abecedou Γ.

Poznamenejme, že obdobná věta platí i pro substituce. Je-li L regulární jazyk nad abecedou Σ a σ je taková substituce z Σ do Γ, že každý z jazyků $\sigma(a)$ pro $a \in \Sigma$ je regulární jazyk nad Γ, pak jazyk $\sigma(L)$ je také regulární jazyk nad Γ.

22.3.0.11 Věta

Třída regulárních jazyků je uzavřena na inversní homomorfismy. Jinými slovy, jestliže h je homomorfismus a L je regulární jazyk nad abecedou Γ, pak jazyk $h^{-1}(L)$ je regulární jazyk nad abecedou Σ.

Připomeňme, že $h^{-1}(L) = \{u \in \Sigma^\star | h(u) \in L\}$.

Příklad: Uvažujme jazyk L nad abecedou $\Gamma = \{a, b\}$ popsaný regulárním výrazem $(\mathbf{00} + \mathbf{1})^\star$ a homomorfismus h, kde $h(a) = 01$ a $h(b) = 10$.

Pak $h^{-1}(L)$ je jazyk nad abecedou $\Sigma = \{a, b\}$ popsaný regulárním výrazem $(\mathbf{ba})^\star$.

22.3.0.12 Reverse

Je dán jazyk L nad abecedou Σ. Pak jazyk L^R definovaný

$$L^R = \left\{ w^R | w \in L \right\}.$$

se nazývá *reverse* jazyka L.

Věta: Třída regulárních jazyků je uzavřena na reverse, přesněji: jestliže L je regulární jazyk nad abecedou Σ, pak je regulární i jazyk L^R.

22.3.0.13 Levý kvocient

Máme dva jazyky L a L_1 nad abecedou Σ. Pak *levý kvocient* je jazyk

$$L_1 \setminus L = \{v | \exists u \in L_1, uv \in L\}.$$

Příklad: Uvažujme jazyky L_1 a L_2 nad abecedou $\Sigma = \{0, 1\}$, kde $L_1 = \left\{0^k 10^n | k, n \geq 0\right\}$, $L_2 = \{10^m 1 | m \geq 0\}$.

Pak $L_2 \setminus L_1 = \emptyset$ a $L_1 \setminus L_2 = \{0^q 1 | q \geq 0\}$.

Věta: Třída regulárních jazyků je uzavřena na levé kvocienty. Přesněji, jestliže L a L_1 jsou regulární jazyky, pak i $L_1 \setminus L$ je regulární jazyk.

22.3.0.14 Pravý kvocient

Máme dva jazyky L a L_2 nad abecedou Σ. Pak *pravý kvocient* je jazyk

$$L/L_2 = \{v | \exists u \in L_2, vu \in L\}.$$

Příklad: Uvažujme jazyky L_1 a L_2 nad abecedou $\Sigma = \{0, 1\}$, kde $L_1 = \left\{0^k 10^n | k, n \geq 0\right\}$, $L_2 = \{10^m 1 | m \geq 0\}$.

Pak $L_2/L_1 = \left\{10^k | k \geq 0\right\}$ a $L_1/L_2 = \emptyset$.

Věta: Třída regulárních jazyků je uzavřena na pravé kvocienty. Přesněji, jestliže L a L_2 jsou regulární jazyky, pak i L/L_2 je regulární jazyk.

22.4 Algoritmická řešitelnost úloh pro regulární jazyky

Pro následující otázky týkající se konečných automatů a jimi přijímaných jazyků existují algoritmy, které dají správnou odpověď.

1. Pro daný konečný automat M (ať deterministický nebo nedeterministický) a slovo $w \in \Sigma^\star$ rozhodnout, zda $w \in L(M)$.
2. Pro daný konečný automat M (ať deterministický nebo nedeterministický) rozhodnout, zda $L(M) \neq \emptyset$.
3. Pro daný konečný automat M rozhodnout, zda $L(M) = \Sigma^\star$.
4. Pro dva konečné automaty M_1 a M_2 rozhodnout, zda $L(M_1) = L(M_2)$.

Tvrzení: Je dán deterministický konečný automat M s n stavy. Pak

1. Jazyk $L(M)$ je neprázdný právě tehdy, když M přijímá slovo w délky $|w| < n$.
2. Jazyk $L(M)$ je nekonečný právě tehdy, když M přijímá slovo v délky $n \leq |v| < 2n$.