Riceove vety

Peter Kostolányi

4. apríla 2017

Na prednáške bolo dokázaných viacero výsledkov o rozhodnuteľnosti resp. nerozhodnuteľnosti vlastností jazykov z tried Chomského hierarchie; príkladmi takýchto vlastností sú prázdnosť, konečnosť, či regulárnosť. *Riceove vety* poskytujú jednotné a všeobecné kritéria, pomocou ktorých možno pre danú vlastnosť *rekurzívne vyčísliteľných* jazykov určiť, či je rozhodnuteľná alebo aspoň rekurzívne vyčísliteľná.

V celom texte týchto poznámok budeme uvažovať iba jazyky nad abecedou $\Sigma = \{0, 1\}$; analogickú teóriu je ale možné vybudovať pre ľubovoľnú pevne danú abecedu Σ.

Vlastnosti rekurzívne vyčísliteľných jazykov

Aby bolo možné vyšetrovať rozhodnuteľnosť vlastností rekurzívne vyčísliteľných jazykov vo všeobecnosti, je nutná aj náležite všeobecná definícia vlastnosti samotnej. Tá sa opiera o jedinú, snáď až príliš očividnú črtu vlastností: každý jazyk buď danú vlastnosť má, alebo danú vlastnosť nemá. Z toho vyplýva, že vlastnosť rekurzívne vyčísliteľných jazykov možno formalizovať jednoducho ako triedu tých rekurzívne vyčísliteľných jazykov, ktoré ju (v intuitívnom ponímaní) majú.

Definícia 1. *Vlastnosť rekurzívne vyčísliteľných jazykov* je ľubovoľná trieda jazykov $\mathscr{S} \subseteq \mathscr{L}_{RE}$.

Príklad 1. Regulárnosť rekurzívne vyčísliteľného jazyka je vlastnosť, ktorú majú všetky regulárne rekurzívne vyčísliteľné jazyky – čiže všetky regulárne jazyky. Formálne preto regulárnosti zodpovedá trieda jazykov $\mathscr{S}_{reg} = \mathscr{R}$.

Prázdnosť rekurzívne vyčísliteľného jazyka je vlastnosť, ktorú má jedine prázdny jazyk $\emptyset$. Formálne preto prázdnosti zodpovedá trieda jazykov $\mathscr{S}_{\emptyset} = \{\emptyset\}$ obsahujúca iba prázdny jazyk. *Pozor:* trieda jazykov $\mathscr{S}_{\emptyset}$ nie je prázdna, ale jednoprvková – obsahuje totiž prázdny jazyk.

Podobne rovnosť so Σ^* je vlastnosť, ktorú má jedine jazyk Σ^*. Formálne preto ide o jednoprvkovú triedu jazykov $\mathscr{S}_{\Sigma^*} = \{\Sigma^*\}$ obsahujúcu iba jazyk Σ^*. $\square$

Ďalej je nutné zaviesť pojem *rozhodnuteľnosti* resp. *rekurzívnej vyčísliteľnosti* vlastnosti rekurzívne vyčísliteľných jazykov. Jazyk sám o sebe je (vo všeobecnosti) nekonečný objekt, každý algoritmus však musí mať konečný vstup. Konečnými popismi rekurzívne vyčísliteľného jazyka sú kódy Turingových strojov, ktoré daný jazyk akceptujú. Pod algoritmom rozhodujúcim vlastnosť $\mathscr{S}$ rekurzívne vyčísliteľných jazykov teda budeme chápať algoritmus, ktorý pre daný kód $\langle A \rangle$ deterministického Turingovho stroja A rozhodne, či $L(A) \in \mathscr{S}$.

Definícia 2. Nech $\mathscr{S} \subseteq \mathscr{L}_{RE}$ je vlastnosť rekurzívne vyčísliteľných jazykov. Vlastnosť $\mathscr{S}$ je *rozhodnuteľná*, ak

$$L_{\mathscr{S}} = \{\langle A \rangle \mid A \text{ je det. TS nad vstupnou abecedou } \{0,1\};\ L(A) \in \mathscr{S}\} \in \mathscr{L}_{rec}.$$

Vlastnosť $\mathscr{S}$ je *rekurzívne vyčísliteľná*, ak

$$L_{\mathscr{S}} = \{\langle A \rangle \mid A \text{ je det. TS nad vstupnou abecedou } \{0,1\};\ L(A) \in \mathscr{S}\} \in \mathscr{L}_{RE}.$$

Poznámka 1. Prechod od jazykov k zodpovedajúcim Turingovým strojom znamená pre vlastnosti prechod od *triedy jazykov* $\mathscr{S}$ k *jazyku* $L_{\mathscr{S}}$. Práve tento prechod nám umožňuje hovoriť o rozhodnuteľnosti resp. čiastočnej rozhodnuteľnosti vlastností, keďže každý rozhodovací problém je z formálneho hľadiska jednoducho jazyk.

Niektoré vlastnosti rekurzívne vyčísliteľných jazykov sú očividne rozhodnuteľné. Uvažujme napríklad vlastnosť $\mathscr{S}$ „rekurzívnej vyčísliteľnosti rekurzívne vyčísliteľného jazyka“. Každý rekurzívne vyčísliteľný jazyk je rekurzívne vyčísliteľný, a teda $\mathscr{S} = \mathscr{L}_{RE}$. Túto vlastnosť teda rozhoduje algoritmus, ktorý pre každý kód $\langle A \rangle$ deterministického Turingovho stroja A ihneď vráti odpoveď „áno“. Vlastnosti rekurzívne vyčísliteľných jazykov, ktoré majú buď všetky rekurzívne vyčísliteľné jazyky, alebo žiaden rekurzívne vyčísliteľný jazyk, nazývame *triviálnymi.*

Definícia 3. Nech $\mathscr{S} \subseteq \mathscr{L}_{RE}$ je vlastnosť rekurzívne vyčísliteľných jazykov. Vlastnosť $\mathscr{S}$ je *triviálna*, ak $\mathscr{S} = \emptyset$ alebo $\mathscr{S} = \mathscr{L}_{RE}$. Vlastnosť $\mathscr{S}$ je *netriviálna*, ak nie je triviálna.

Poznámka 2. Rovnosť $\mathscr{S} = \emptyset$ znamená – keďže $\mathscr{S}$ je trieda jazykov – že žiaden jazyk nemá vlastnosť $\mathscr{S}$. Táto rovnosť nemá spoločné *vôbec nič* s problémom prázdnosti jazyka.

Tvrdenie 1. *Nech $\mathscr{S} \subseteq \mathscr{L}_{RE}$ je* triviálna *vlastnosť rekurzívne vyčísliteľných jazykov. Potom je vlastnosť $\mathscr{S}$ rozhodnuteľná.*

Dôkaz. Ak $\mathscr{S} = \emptyset$, algoritmus rozhodujúci $\mathscr{S}$ vráti na každom vstupe „nie". Ak $\mathscr{S} = \mathscr{L}_{RE}$, algoritmus rozhodujúci $\mathscr{S}$ vráti na každom vstupe „áno". □

O chvíľu dokážeme prvú Riceovu vetu, ktorá hovorí, že triviálne vlastnosti sú *jediné* rozhodnuteľné vlastnosti rekurzívne vyčísliteľných jazykov.

Poznámka 3. Čitateľ by sa mal uistiť, že je mu jasný rozdiel medzi úlohami, ktoré v teórii vlastností rekurzívne vyčísliteľných jazykov zohrávajú jazyky a triedy jazykov. Dôkladné uvedomenie si tohto rozdielu je nutnou (a v zásade takmer aj postačujúcou) podmienkou pochopenia Riceových viet.

Prvá Riceova veta

Prv, než vyslovíme a dokážeme prvú Riceovu vetu, dokážeme nerozhodnuteľnosť jednej konkrétnej vlastnosti rekurzívne vyčísliteľných jazykov. Priamočiarym zovšeobecnením použitej konštrukcie potom dostaneme priamo dôkaz prvej Riceovej vety.

Z hľadiska dôkazových techník nepôjde o nič prevratné. Všetky metódy používané v rámci tohto oddielu sme už používali – čitateľ si môže osviežiť pamäť napríklad v poznámkach k rozhodnuteľnosti zo zimného semestra.

Príklad 2. Dokážeme, že rovnosť s jazykom Σ^* je pre rekurzívne vyčísliteľné jazyky nerozhodnuteľná.[1] To znamená dokázať nerozhodnuteľnosť problému daného nasledovne:

Vstup: Kód $\langle A \rangle$ deterministického Turingovho stroja A nad vstupnou abecedou $\{0,1\}$.

Výstup: „Áno" práve vtedy, keď $L(A) = \Sigma^*$.

Nami skúmaná vlastnosť rekurzívne vyčísliteľných jazykov je z formálneho hľadiska trieda jazykov $\mathscr{S}_{\Sigma^*} = \{\Sigma^*\}$. Opísanému rozhodovaciemu problému pritom zodpovedá jazyk

$$L_{\mathscr{S}_{\Sigma^*}} = \{\langle A \rangle \mid A \text{ je det. TS nad vstupnou abecedou } \{0,1\};\ L(A) \in \mathscr{S}_{\Sigma^*}\}.$$

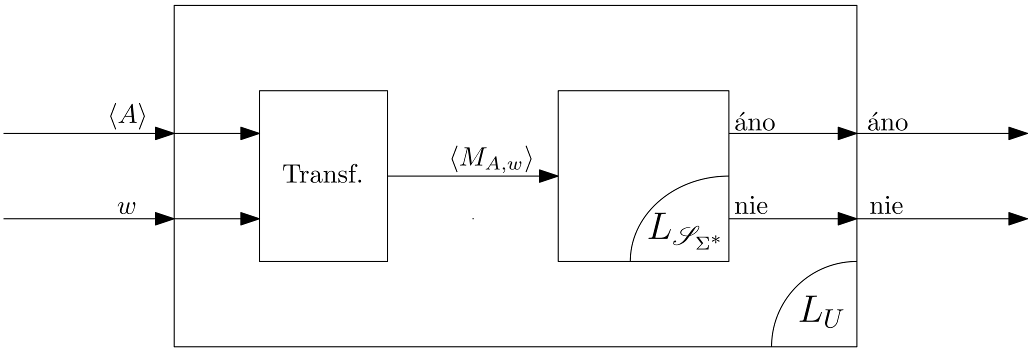
Redukciou univerzálneho problému na náš problém dokážeme, že vlastnosť $\mathscr{S}_{\Sigma^*}$ nie je rozhodnuteľná. Za účelom sporu predpokladajme, že je rozhodnuteľná – teda existuje deterministický Turingov stroj, ktorý sa na každom vstupe zastaví a ktorý akceptuje $L_{\mathscr{S}_{\Sigma^*}}$.

Ukážeme, že v takom prípade možno skonštruovať stroj rozhodujúci univerzálny problém. Vstupom univerzálneho problému je kód $\langle A \rangle$ deterministického Turingovho stroja A a slovo w. Dvojicu vstupov $\langle A \rangle, w$ pritom treba akceptovať práve vtedy, keď $w \in L(A)$. Na rozhodovanie univerzálneho problému preto očividne stačí vedieť transformovať vstupy $\langle A \rangle, w$ na kód $\langle M_{A,w} \rangle$ deterministického Turingovho stroja $M_{A,w}$ takého, že platí

$$L(M_{A,w}) = \Sigma^* \text{ práve vtedy, keď } w \in L(A). \tag{1}$$

Stroj rozhodujúci univerzálny problém totiž môže pracovať tak, že transformuje svoj vstup $\langle A \rangle, w$ na $\langle M_{A,w} \rangle$ a na tomto kóde spustí stroj rozhodujúci vlastnosť $\mathscr{S}_{\Sigma^*}$. Ak je výstupom tohto výpočtu „áno", platí $L(M_{A,w}) = \Sigma^*$, a teda podľa (1) aj $w \in L(A)$. Stroj pre univerzálny problém tak tiež môže vrátiť na výstupe „áno". Ak je naopak výstupom výpočtu stroja pre $\mathscr{S}_{\Sigma^*}$ odpoveď „nie", nutne $L(M_{A,w}) \neq \Sigma^*$, a teda podľa (1) dostávame $w \notin L(A)$. Stroj pre univerzálny problém preto môže vrátiť odpoveď „nie". Schéma tejto redukcie je na obrázku 1.

[1] V celom texte týchto poznámok predpokladáme $\Sigma = \{0,1\}$; analogické tvrdenie je ale možné dokázať pre ľubovoľnú pevne danú abecedu.

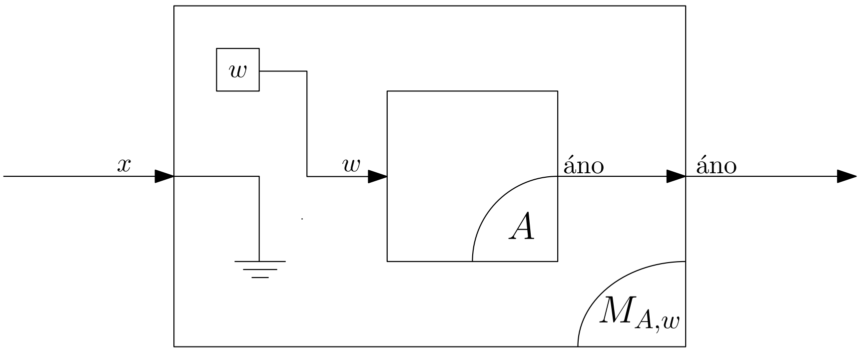


Obr. 1: Schéma redukcie univerzálneho problému na problém zodpovedajúci vlastnosti $\mathscr{S}_{\Sigma^*}$.

Stroj $M_{A,w}$ skonštruujeme tak, aby platilo

$$L(M_{A,w}) = \begin{cases} \Sigma^* & \text{ak } w \in L(A) \\ \emptyset & \text{inak} \end{cases},$$

čo je očividne silnejšia podmienka, než (1). Pracovať bude tak, že svoj vstup x ihneď zahodí a nahradí ho slovom w, ktoré má „pevne zadrôtované v prechodovej funkcii“. Následne spustí simuláciu výpočtu stroja A (ktorý má tiež „pevne zadrôtovaný v prechodovej funkcii“) na slove w a akceptuje práve vtedy, keď akceptuje stroj A. Schéma konštrukcie stroja $M_{A,w}$ je na obrázku 2.



Obr. 2: Schematické znázornenie konštrukcie stroja $M_{A,w}$.

Transformáciu vstupov $\langle A\rangle, w$ na kód $\langle M_{A,w}\rangle$ stroja $M_{A,w}$ možno očividne realizovať algoritmicky. Tvrdenie je teda dokázané. □

Veta 1 (prvá Riceova)**.** *Nech $\mathscr{S} \subseteq \mathscr{L}_{RE}$ je* netriviálna *vlastnosť rekurzívne vyčísliteľných jazykov. Potom je vlastnosť $\mathscr{S}$ nerozhodnuteľná.*

Myšlienka dôkazu. Bez ujmy na všeobecnosti môžeme predpokladať, že prázdny jazyk nemá vlastnosť $\mathscr{S}$ – čiže $\emptyset \notin \mathscr{S}$. V prípade $\emptyset \in \mathscr{S}$ totiž stačí uvažovať vlastnosť $\mathscr{S}^C$.

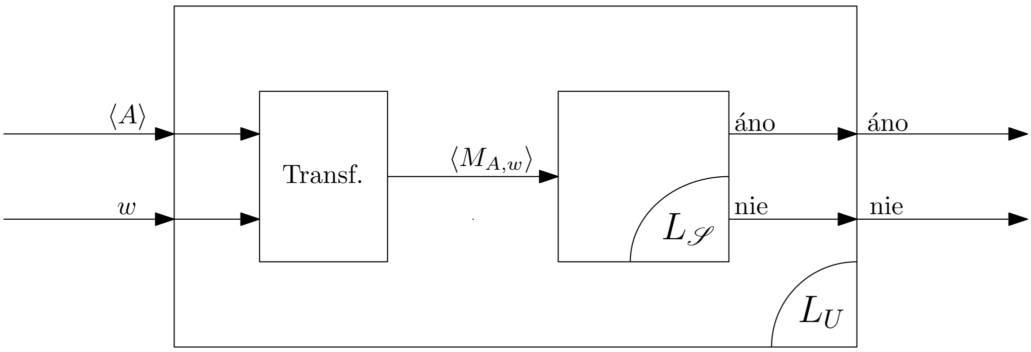
Zovšeobecníme postup z predchádzajúceho príkladu; budeme teda postupovať redukciou z univerzálneho problému. Jeho vstupy $\langle A\rangle, w$ využijeme na konštrukciu vhodného stroja $M_{A,w}$, na kóde ktorého spustíme (hypotetický) stroj rozhodujúci vlastnosť $\mathscr{S}$. Pre stroj $M_{A,w}$ bude platiť: ak $w \notin L(A)$, tak $L(M_{A,w}) = \emptyset \notin \mathscr{S}$. Rozdiel oproti príkladu 2 nastane pre $w \in L(A)$. Tu už totiž vo všeobecnosti nefunguje voľba $L(M_{A,w}) = \Sigma^*$, keďže nemusí platiť $\Sigma^* \in \mathscr{S}$. Stroj $M_{A,w}$ preto treba skonštruovať tak, aby v prípade $w \in L(A)$ platilo $L(M_{A,w}) = L$, kde L je ľubovoľný jazyk v $\mathscr{S}$. Používajúc teda označenia z príkladu 2: ak simulácia stroja A na slove w prebehne úspešne (stroj A akceptuje), treba ešte spustiť simuláciu stroja pre jazyk L na slove x. Až po akceptovaní slova x strojom pre L ho môže akceptovať aj stroj $M_{A,w}$. □

Dôkaz. Bez ujmy na všeobecnosti, nech $\emptyset \notin \mathscr{S}$. Redukciou z univerzálneho problému dokážeme, že vlastnosť $\mathscr{S}$ nie je rozhodnuteľná. Za účelom sporu predpokladajme opak – nech teda existuje deterministický Turingov stroj, ktorý sa na každom vstupe zastaví a ktorý akceptuje $L_{\mathscr{S}}$.

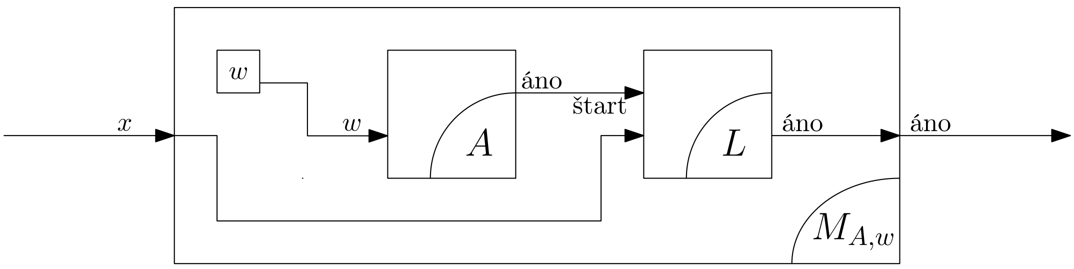
Dokážeme, že potom možno skonštruovať stroj rozhodujúci univerzálny problém. Vstupom univerzálneho problému je kód $\langle A \rangle$ deterministického Turingovho stroja A a slovo w, pričom dvojicu vstupov $\langle A \rangle, w$ treba akceptovať práve vtedy, keď $w \in L(A)$. Na rozhodovanie univerzálneho problému preto stačí vedieť transformovať vstupy $\langle A \rangle, w$ na kód $\langle M_{A,w} \rangle$ deterministického Turingovho stroja $M_{A,w}$ takého, že platí

$$L(M_{A,w}) \in \mathscr{S} \text{ práve vtedy, keď } w \in L(A). \tag{2}$$

Stroj rozhodujúci univerzálny problém totiž môže pracovať tak, že najprv transformuje svoj vstup $\langle A \rangle, w$ na $\langle M_{A,w} \rangle$, pričom na tomto kóde spustí stroj rozhodujúci vlastnosť $\mathscr{S}$. Ak je výstupom tohto výpočtu „áno“, platí $L(M_{A,w}) \in \mathscr{S}$, a teda podľa (2) aj $w \in L(A)$. Ak je výstupom výpočtu stroja pre $\mathscr{S}$ odpoveď „nie“, nutne $L(M_{A,w}) \notin \mathscr{S}$, a teda podľa (2) dostávame $w \notin L(A)$. Schéma tejto redukcie je na obrázku 3.



Obr. 3: Schéma redukcie univerzálneho problému na problém zodpovedajúci vlastnosti $\mathscr{S}$.



Obr. 4: Schematické znázornenie konštrukcie stroja $M_{A,w}$.

Nech $L \in \mathscr{S}$ je ľubovoľný rekurzívne vyčísliteľný jazyk s vlastnosťou $\mathscr{S}$ – keďže $\emptyset \notin \mathscr{S}$ a $\mathscr{S}$ je netriviálna, nutne existuje aspoň jeden taký jazyk L. Stroj $M_{A,w}$ skonštruujeme tak, aby platilo

$$L(M_{A,w}) = \begin{cases} L & \text{ak } w \in L(A) \\ \emptyset & \text{inak} \end{cases}$$

– to je silnejšia podmienka, než (2). Pracovať bude tak, že najprv spustí simuláciu výpočtu stroja A na slove w, kde A aj w sú „pevne zadrôtované v prechodovej funkcii“. Ak stroj A akceptuje, spustí ešte simuláciu stroja pre L (tiež „pevne zadrôtovaného v prechodovej funkcii“) na vstupe x a akceptuje, ak akceptuje aj ten. Ak stroj A alebo stroj pre L neakceptuje, neakceptuje ani stroj $M_{A,w}$. Schéma konštrukcie stroja $M_{A,w}$ je na obrázku 4.

Transformáciu vstupov $\langle A \rangle, w$ na kód $\langle M_{A,w} \rangle$ stroja $M_{A,w}$ možno očividne realizovať algoritmicky. Veta je teda dokázaná. $\square$

Použitie prvej Riceovej vety

Nech $\mathscr{S}$ je ľubovoľná vlastnosť rekurzívne vyčísliteľných jazykov. Použitie prvej Riceovej vety na vyšetrenie rozhodnuteľnosti vlastnosti $\mathscr{S}$ je pomerne priamočiarou záležitosťou:

- V prípade, že existuje aspoň jeden jazyk $L_1 \in \mathscr{L}_{RE}$, ktorý vlastnosť $\mathscr{S}$ má a aspoň jeden jazyk $L_2 \in \mathscr{L}_{RE}$, ktorý vlastnosť $\mathscr{S}$ nemá, je vlastnosť $\mathscr{S}$ nerozhodnuteľná.
- V opačnom prípade je vlastnosť $\mathscr{S}$ triviálna, a teda rozhodnuteľná.

Príklad 3. Uvažujme napríklad vlastnosť $\mathscr{S}_1$ bezkontextovosti rekurzívne vyčísliteľných jazykov. Každý bezkontextový jazyk má vlastnosť $\mathscr{S}_1$, ale napríklad jazyk $\{ww \mid w \in \{0,1\}^*\} \in \mathscr{L}_{RE}$ vlastnosť $\mathscr{S}_1$ nemá. Preto je vlastnosť $\mathscr{S}_1$ nerozhodnuteľná.

Analogicky možno dokázať, že napríklad prázdnosť, regulárnosť, či rovnosť so Σ^* sú nerozhodnuteľné vlastnosti.

Vezmime teraz vlastnosť $\mathscr{S}_2$, ktorú majú všetky rekurzívne vyčísliteľné jazyky L také, že $L^5 \in \mathscr{L}_{RE}$. Potom ale zrejme $\mathscr{S}_2 = \mathscr{L}_{RE}$, keďže neexistuje žiaden rekurzívne vyčísliteľný jazyk, pre ktorý táto vlastnosť neplatí. Preto je vlastnosť $\mathscr{S}_2$ triviálna, a teda rozhodnuteľná.

Poznámka 4. Pri použití prvej Riceovej vety je potrebné mať na zreteli predovšetkým to, či ide o rozhodovanie vlastnosti rekurzívne vyčísliteľných *jazykov*. Uvažujme napríklad rozhodovací problém, v ktorom treba pre daný kód $\langle A \rangle$ Turingovho stroja A určiť, či stroj A urobí na vstupe ε aspoň tri kroky. Táto vlastnosť je očividne rozhodnuteľná, keďže stačí pomocou univerzálneho stroja odsimulovať najviac tri kroky stroja A na vstupe ε. Na dôkaz tejto skutočnosti však nie je možné použiť prvú Riceovu vetu, keďže nejde o rozhodovanie vlastnosti jazyka $L(A)$, ale vlastnosti konkrétneho Turingovho stroja A.

Druhá Riceova veta

Druhá Riceova veta poskytuje kritérium rekurzívnej vyčísliteľnosti vlastností rekurzívne vyčísliteľných jazykov. Než túto vetu sformulujeme, dokážeme o dvoch konkrétnych vlastnostiach, že nie sú rekurzívne vyčísliteľné. Použité techniky neskôr využijeme pri dôkaze druhej Riceovej vety.

Príklad 4. Dokážeme, že rekurzívnosť rekurzívne vyčísliteľných jazykov nie je rekurzívne vyčísliteľná – teda, že nie je rekurzívne vyčísliteľný nasledujúci rozhodovací problém:

Vstup: Kód $\langle A \rangle$ deterministického Turingovho stroja A nad vstupnou abecedou $\{0,1\}$.

Výstup: „Áno“ práve vtedy, keď $L(A) \in \mathscr{L}_{rec}$.

Takáto vlastnosť je z formálneho hľadiska trieda jazykov $\mathscr{S} = \mathscr{L}_{rec}$. Opísanému rozhodovaciemu problému potom zodpovedá jazyk

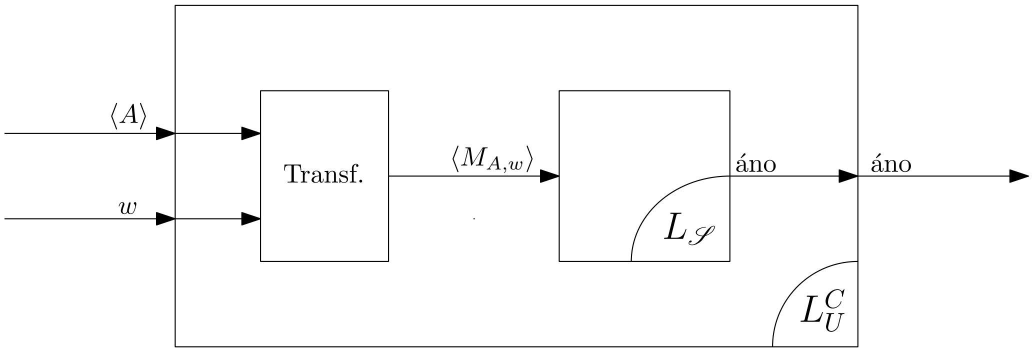
$$L_{\mathscr{S}} = \{\langle A \rangle \mid A \text{ je det. TS nad vstupnou abecedou } \{0,1\};\ L(A) \in \mathscr{S}\}.$$

Za účelom sporu predpokladajme, že je vlastnosť $\mathscr{S}$ rekurzívne vyčísliteľná, a teda existuje deterministický Turingov stroj akceptujúci jazyk $L_{\mathscr{S}}$.

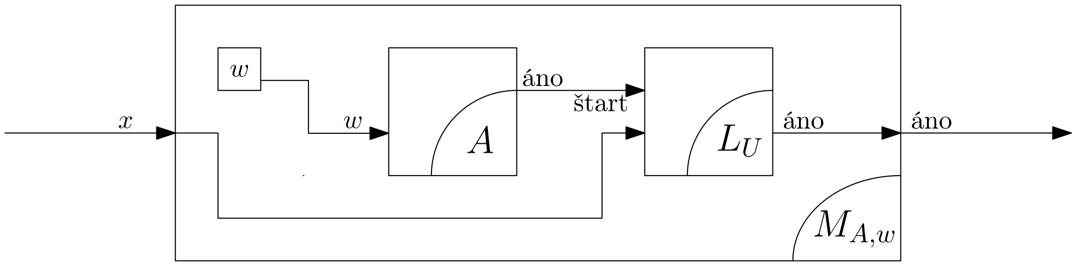
Ukážeme, že potom existuje deterministický Turingov stroj akceptujúci jazyk L_U^C (spor). Tento jazyk obsahuje neplatné vstupy univerzálneho problému (teda reťazce, ktoré nemožno interpretovať ako dvojicu „kód Turingovho stroja – vstup“) a reťazce reprezentujúce kód $\langle A \rangle$ deterministického Turingovho stroja A spolu so slovom w takým, že $w \notin L(A)$. Neplatné vstupy možno identifikovať pomerne jednoducho, preto sa stačí sústrediť na zostávajúci prípad – Turingov stroj akceptujúci L_U^C existuje práve vtedy, keď existuje Turingov stroj, ktorý pre daný kód $\langle A \rangle$ deterministického Turingovho stroja A a dané slovo w akceptuje práve vtedy, keď $w \notin L(A)$.

Na konštrukciu takéhoto stroja je zjavne postačujúce vedieť transformovať transformovať vstupy $\langle A \rangle, w$ na kód $\langle M_{A,w} \rangle$ deterministického Turingovho stroja $M_{A,w}$, pre ktorý platí

$$L(M_{A,w}) \in \mathscr{L}_{rec} \text{ práve vtedy, keď } w \notin L(A). \tag{3}$$



Obr. 5: Schéma redukcie komplementárneho univerzálneho problému na problém rekurzívnosti za predpokladu, že vstupy stroja pre L_U^C sú platnými vstupmi univerzálneho problému. Do jazyka L_U^C patria navyše aj všetky neplatné vstupy univerzálneho problému.



Obr. 6: Schematické znázornenie konštrukcie stroja $M_{A,w}$.

Stroj pre L_U^C totiž môže pracovať tak, že vstupy $\langle A\rangle, w$ prerobí na kód $\langle M_{A,w}\rangle$, na ktorom spustí stroj pre rekurzívnosť. Ak tento stroj akceptuje, musí podľa (3) platiť $w \notin L(A)$, a teda môže akceptovať aj stroj pre L_U^C. Analogicky pre opačnú implikáciu. Schéma redukcie je na obrázku 5.

Stroj $M_{A,w}$ skonštruujeme tak, aby platilo

$$L(M_{A,w}) = \begin{cases} \emptyset & \text{ak } w \notin L(A) \\ L_U & \text{inak} \end{cases} .$$

Táto podmienka je silnejšia, než (3) – pre $w \notin L(A)$ je jazyk $L(M_{A,w})$ prázdny, a teda rekurzívny, kým pre $w \in L(A)$ platí $L(M_{A,w}) = L_U$, čo nie je rekurzívny jazyk. Stroj $M_{A,w}$ môže na vstupe x pracovať tak, že najprv na „pevne zadrôtovanom“ slove w spustí simuláciu „pevne zadrôtovaného“ stroja A. Ak stroj A akceptuje, spustí sa simulácia univerzálneho stroja na vstupe x a stroj $M_{A,w}$ akceptuje práve vtedy, keď akceptuje univerzálny stroj. Ak stroj A neakceptuje, neakceptuje ani stroj $M_{A,w}$. Schéma konštrukcie stroja $M_{A,w}$ je na obrázku 6.

Transformáciu $\langle A\rangle, w$ na $\langle M_{A,w}\rangle$ možno zjavne realizovať algoritmicky. $\square$

Cvičenie 1. Dokážte, že komplementárna vlastnosť k rekurzívnosti – vlastnosť $\mathscr{S} = \mathscr{L}_{RE} - \mathscr{L}_{rec}$ – tiež nie je rekurzívne vyčísliteľná.

Príklad 5. Dokážeme, že nekonečnosť rekurzívne vyčísliteľných jazykov nie je rekurzívne vyčísliteľná – teda, že nie je rekurzívne vyčísliteľný nasledujúci rozhodovací problém:

Vstup: Kód $\langle A\rangle$ deterministického Turingovho stroja A nad vstupnou abecedou $\{0,1\}$.

Výstup: „Áno“ práve vtedy, keď je jazyk $L(A)$ nekonečný.

Nekonečnosť rekurzívne vyčísliteľných jazykov je z formálneho hľadiska trieda jazykov $\mathscr{S}$ pozostávajúca z nekonečných jazykov v $\mathscr{L}_{RE}$. Opísanému rozhodovaciemu problému zodpovedá jazyk

$$L_{\mathscr{S}} = \{\langle A\rangle \mid A \text{ je det. TS nad vstupnou abecedou } \{0,1\};\ L(A) \in \mathscr{S}\}.$$

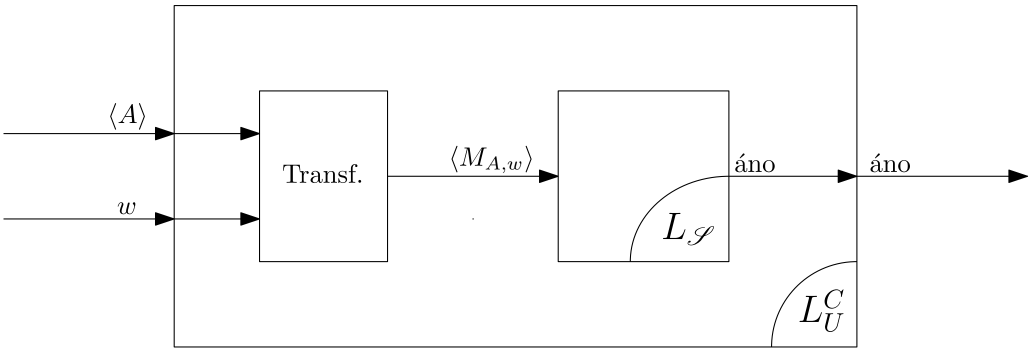
Za účelom sporu predpokladajme, že je vlastnosť $\mathscr{S}$ rekurzívne vyčísliteľná, a teda existuje deterministický Turingov stroj akceptujúci jazyk $L_{\mathscr{S}}$.

Ukážeme, že potom existuje deterministický Turingov stroj akceptujúci jazyk L_U^C (spor). Argumentácia z príkladu 4 ukazuje, že je postačujúce zamerať sa na platné vstupy univerzálneho problému a ukázať, že existuje deterministický Turingov stroj, ktorý daný kód $\langle A\rangle$ deterministického Turingovho stroja A a slovo w akceptuje práve vtedy, keď $w \notin L(A)$.

Takýto stroj by zjavne existoval, keby sme vedeli transformovať vstupy $\langle A\rangle, w$ na kód $\langle M_{A,w}\rangle$ deterministického Turingovho stroja $M_{A,w}$, pre ktorý platí

$$L(M_{A,w}) \text{ je nekonečný práve vtedy, keď } w \notin L(A). \tag{4}$$

Stroj pre L_U^C by potom mohol jednoducho transformovať vstupy $\langle A\rangle, w$ na $\langle M_{A,w}\rangle$ a na tomto kóde spustiť stroj pre nekonečnosť. Schéma redukcie je na obrázku 7.



Obr. 7: Schéma redukcie komplementárneho univerzálneho problému na problém nekonečnosti za predpokladu, že vstupy stroja pre L_U^C sú platnými vstupmi univerzálneho problému. Do jazyka L_U^C patria navyše aj všetky neplatné vstupy univerzálneho problému.

Nech L je ľubovoľný nekonečný rekurzívne vyčísliteľný jazyk a nech A_L je deterministický Turingov stroj akceptujúci L, ktorý na vstupe dĺžky n urobí pred akceptáciou aspoň n krokov výpočtu (zrejme ide o normálny tvar). Stroj $M_{A,w}$ na každom vstupe x najprv odsimuluje výpočet stroja A_L, pričom vo výpočte pokračuje iba ak stroj A_L akceptuje. V takom prípade sa spustí simulácia $|x|$ krokov výpočtu stroja A na vstupe w. Ak stroj A počas simulovaných $|x|$ krokov výpočtu slovo w neakceptuje, stroj $M_{A,w}$ akceptuje slovo x.

Ak teda $w \notin L(A)$, stroj $M_{A,w}$ akceptuje práve všetky slová z jazyka L – slová mimo L sa „odfiltrujú“ simuláciou stroja A_L; slová x z L sa ale akceptujú, pretože stroj A slovo w neakceptuje na žiaden počet krokov, a teda ani na $|x|$ krokov. Ak naopak $w \in L(A)$, stroj $M_{A,w}$ akceptuje iba také slová x z jazyka L, že stroj A slovo w na $|x|$ krokov ešte neakceptuje. Takýchto slov je ale vďaka normálnemu tvaru stroja A_L iba konečne veľa. Platí teda

$$L(M_{A,w}) = \begin{cases} L & \text{ak } w \notin L(A) \\ \text{nejaký konečný podjazyk jazyka } L & \text{inak} \end{cases},$$

čo je očividne silnejšie, než (4). Schéma konštrukcie stroja $M_{A,w}$ je na obrázku 8.

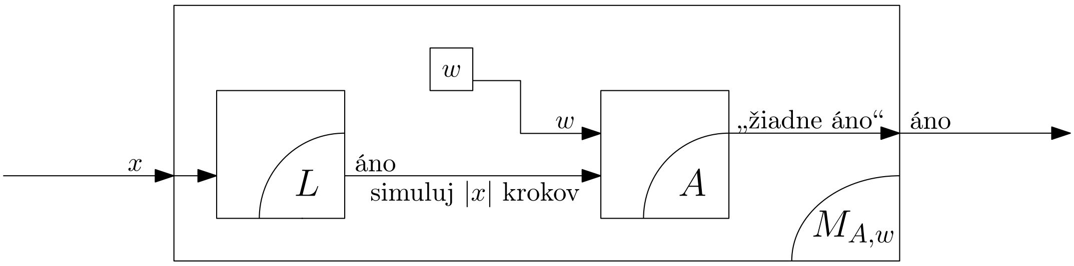
Keďže možno transformáciu $\langle A\rangle, w$ na $\langle M_{A,w}\rangle$ očividne realizovať algoritmicky, tvrdenie je dokázané. $\square$

Prv, než sformulujeme druhú Riceovu vetu, zavedieme v jej znení sa vyskytujúci pojem rekurzívnej vyčísliteľnosti množiny[2] konečných jazykov.

Definícia 4. Nech $\mathscr{M}$ je ľubovoľná množina konečných jazykov nad abecedou Σ. Hovoríme, že množina $\mathscr{M}$ je rekurzívne vyčísliteľná, ak je rekurzívne vyčísliteľný jazyk

$$L_{\mathscr{M}} = \{w_1\#w_2\#\ldots\#w_n \mid n \in \mathbb{N}; \{w_1, w_2, \ldots, w_n\} \in \mathscr{M}\}.$$

[2] Pôjde naozaj o množiny, keďže jazyky v nich budú nad pevne danou abecedou Σ.



Obr. 8: Schematické znázornenie konštrukcie stroja $M_{A,w}$.

Každý konečný jazyk sa teda jednoznačne zakóduje do reťazca a (vo všeobecnosti nekonečný) jazyk takýchto reťazcov $L_{\mathscr{M}}$ musí byť rekurzívne vyčísliteľný.

V rámci týchto poznámok pracujeme výhradne s jazykmi nad abecedou $\Sigma = \{0, 1\}$. Tak treba chápať aj nasledujúcu formuláciu druhej Riceovej vety.

Veta 2 (druhá Riceova)**.** *Nech $\mathscr{S} \subseteq \mathscr{L}_{RE}$ je vlastnosť rekurzívne vyčísliteľných jazykov. Vlastnosť $\mathscr{S}$ je rekurzívne vyčísliteľná práve vtedy, keď sú splnené nasledujúce podmienky:*

(i) *Pre všetky $L, L' \in \mathscr{L}_{RE}$ také, že $L \in \mathscr{S}$ a $L' \supseteq L$ platí $L' \in \mathscr{S}$.*

(ii) *Pre všetky $L \in \mathscr{S}$ existuje konečný jazyk $L_0 \subseteq L$ taký, že $L_0 \in \mathscr{S}$.*

(iii) *Množina všetkých konečných jazykov v $\mathscr{S}$ je rekurzívne vyčísliteľná.*

Dôkaz. $\Rightarrow$: Dokážeme, že ak je $\mathscr{S}$ rekurzívne vyčísliteľná, sú splnené podmienky (i) až (iii).

(i) Nepriamo. Ukážeme, že ak neplatí (i), vlastnosť $\mathscr{S}$ nie je rekurzívne vyčísliteľná.

Za účelom sporu predpokladajme, že vlastnosť $\mathscr{S}$ je rekurzívne vyčísliteľná. Ukážeme, že v takom prípade $L_U^C \in \mathscr{L}_{RE}$ (spor).

Postup, ktorý použijeme na dôkaz tohto tvrdenia, bude priamočiarym zovšeobecnením postupu z príkladu 4. V príklade 4 platilo $\mathscr{S} = \mathscr{L}_{rec}$ a pre každú dvojicu vstupov $\langle A\rangle, w$ sme konštruovali Turingov stroj $M_{A,w}$ taký, že $L(M_{A,w}) \in \mathscr{S}$ práve vtedy, keď $w \notin L(A)$. Presnejšie, pre stroj $M_{A,w}$ z príkladu 4 platilo

$$L(M_{A,w}) = \begin{cases} \emptyset & \text{ak } w \notin L(A) \\ L_U & \text{inak} \end{cases},$$

čo bolo plne vyhovujúce, keďže pre $\mathscr{S} = \mathscr{L}_{rec}$ máme $\emptyset \in \mathscr{S}$ a $L_U \notin \mathscr{S}$.

Jediná prekážka, ktorá nám zabraňuje použiť navlas rovnakú redukciu aj v tomto prípade, je práve skutočnosť, že nemáme zaručené $\emptyset \in \mathscr{S}$ a $L_U \notin \mathscr{S}$. Keďže ale neplatí (i), určite existuje dvojica jazykov $L, L' \in \mathscr{L}_{RE}$ takých, že $L \subseteq L'$, $L \in \mathscr{S}$ a $L' \notin \mathscr{S}$. V nasledujúcom ukážeme, že je možné skonštruovať Turingov stroj, pre ktorý platí

$$L(M_{A,w}) = \begin{cases} L & \text{ak } w \notin L(A) \\ L' & \text{inak} \end{cases}.$$

Zvyšok dôkazu bude úplne rovnaký ako v príklade 4.

Pre úplnosť uvedieme kompletný dôkaz. Predpokladáme, že neplatí (i) a $\mathscr{S}$ je rekurzívne vyčísliteľná. Chceme ukázať, že v takom prípade $L_U^C \in \mathscr{L}_{RE}$ (spor). Ak sa, podobne ako v príklade 4, obmedzíme na platné vstupy univerzálneho problému, zjavne stačí vedieť transformovať vstupy $\langle A\rangle, w$ na kód $\langle M_{A,w}\rangle$ deterministického Turingovho stroja $M_{A,w}$, pre ktorý platí

$$L(M_{A,w}) \in \mathscr{S} \text{ práve vtedy, keď } w \notin L(A). \tag{5}$$

Stroj akceptujúci L_U^C totiž môže pracovať tak, že vstupy $\langle A\rangle, w$ prerobí na kód $\langle M_{A,w}\rangle$, na ktorom spustí stroj pre vlastnosť $\mathscr{S}$. Ak tento stroj akceptuje, podľa (5) musí platiť $w \notin L(A)$, a teda môže akceptovať aj stroj pre L_U^C. Analogicky pre opačnú implikáciu. Schéma redukcie je na obrázku 9.

Nech $L, L' \in \mathscr{L}_{RE}$ sú jazyky také, že $L \subseteq L'$, $L \in \mathscr{S}$ a $L' \notin \mathscr{S}$ – takáto dvojica jazykov musí existovať, keďže neplatí (i). Stroj $M_{A,w}$ skonštruujeme tak, aby platilo

$$L(M_{A,w}) = \begin{cases} L & \text{ak } w \notin L(A) \\ L' & \text{inak} \end{cases} .$$

Táto podmienka je očividne silnejšia, než (5). Stroj $M_{A,w}$ bude na vstupe x pracovať tak, že najprv na „pevne zadrôtovanom“ slove w spustí simuláciu „pevne zadrôtovaného“ stroja A a súčasne spustí simuláciu stroja pre L na slove x, pričom kroky jednotlivých strojov bude vykonávať striedavo po jednom. Ak stroj A akceptuje, spustí sa navyše simulácia stroja pre L' na vstupe x. Stroj $M_{A,w}$ akceptuje práve vtedy, keď akceptuje stroj pre L alebo stroj pre L' (ak bol spustený). V prípade, že stroj A neakceptuje slovo w, stroj $M_{A,w}$ zjavne akceptuje jazyk L. Ak naopak stroj A slovo w akceptuje, stroj $M_{A,w}$ akceptuje jazyk $L \cup L' = L'$. Schéma konštrukcie stroja $M_{A,w}$ je na obrázku 10.

Transformáciu vstupov $\langle A\rangle, w$ na kód $\langle M_{A,w}\rangle$ stroja $M_{A,w}$ možno zjavne realizovať algoritmicky, a preto je táto časť tvrdenia dokázaná.

(ii) Nepriamo. Ukážeme, že ak neplatí (ii), vlastnosť $\mathscr{S}$ nie je rekurzívne vyčísliteľná.

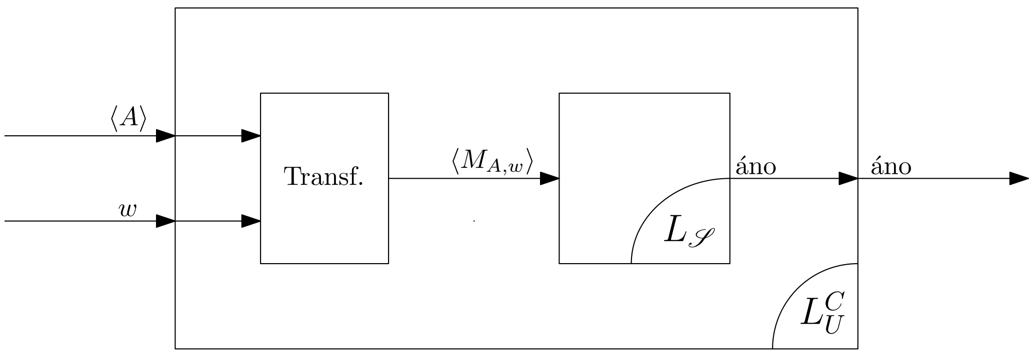
Za účelom sporu predpokladajme, že je vlastnosť $\mathscr{S}$ rekurzívne vyčísliteľná. Ukážeme, že v takom prípade $L_U^C \in \mathscr{L}_{RE}$ (spor).

Tvrdenie dokážeme priamočiarym zovšeobecnením postupu z príkladu 5, kde sme dokázali, že nekonečnosť rekurzívne vyčísliteľných jazykov nie je rekurzívne vyčísliteľná. Pre každú dvojicu vstupov $\langle A\rangle, w$ sme tam konštruovali deterministický Turingov stroj $M_{A,w}$ taký, že $L(M_{A,w})$ je nekonečný práve vtedy, keď $w \notin L(A)$. Presnejšie, pre Turingov stroj $M_{A,w}$ platilo

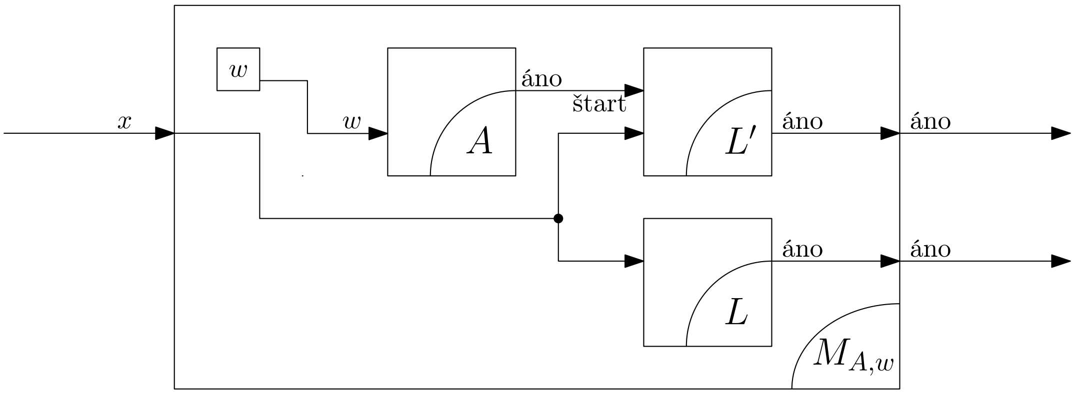
$$L(M_{A,w}) = \begin{cases} L & \text{ak } w \notin L(A) \\ \text{nejaký konečný podjazyk jazyka } L & \text{inak} \end{cases} ,$$

kde L je nejaký pevne daný nekonečný rekurzívne vyčísliteľný jazyk.

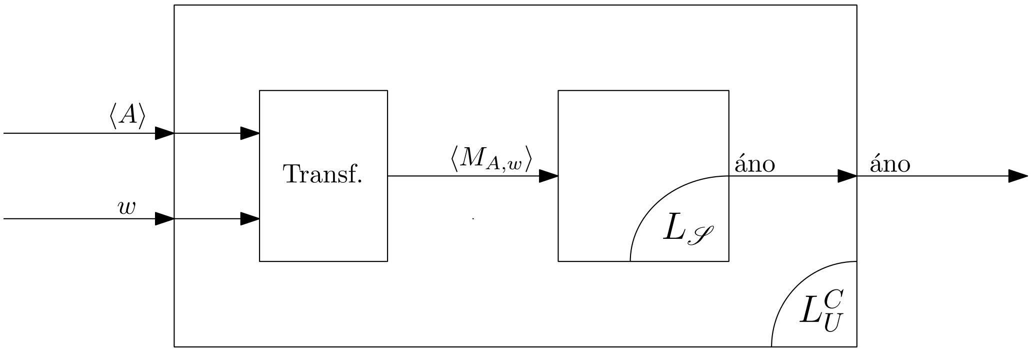
Bezo zmeny túto redukciu použiť nemôžeme, pretože v našom prípade už vo všeobecnosti neplatí $L \in \mathscr{S}$ a nie je ani zaručené, že konečné podjazyky jazyka L vlastnosť $\mathscr{S}$ nemajú. Za predpokladu, že neplatí (ii) však nutne existuje *jeden konkrétny* jazyk $L \in \mathscr{S}$ taký, že žiaden jeho konečný podjazyk L_0 nemá vlastnosť $\mathscr{S}$. Jazyk L je navyše



Obr. 9: Schéma redukcie komplementárneho univerzálneho problému na problém rozhodovania vlastnosti $\mathscr{S}$ za predpokladu, že vstupy stroja pre L_U^C sú platnými vstupmi univerzálneho problému. Do jazyka L_U^C patria navyše aj všetky neplatné vstupy univerzálneho problému.



Obr. 10: Schematické znázornenie konštrukcie stroja $M_{A,w}$.



Obr. 11: Schéma redukcie komplementárneho univerzálneho problému na problém rozhodovania vlastnosti $\mathscr{S}$ za predpokladu, že vstupy stroja pre L_U^C sú platnými vstupmi univerzálneho problému. Do jazyka L_U^C patria navyše aj všetky neplatné vstupy univerzálneho problému.

nutne nekonečný, pretože inak by bolo možné ako konečný podjazyk jazyka L s vlastnosťou $\mathscr{S}$ vziať $L_0 = L$. Ak teda upravíme redukciu z príkladu 5 tak, aby v nej stroj $M_{A,w}$ použil práve tento jazyk L, bude zvyšok dôkazu úplne rovnaký.

Pre úplnosť ešte uvedieme kompletný dôkaz. Nech neplatí (ii) a vlastnosť $\mathscr{S}$ je rekurzívne vyčísliteľná. Zostrojíme deterministický Turingov stroj akceptujúci jazyk L_U^C (spor).

Za predpokladu rekurzívnej vyčísliteľnosti vlastnosti $\mathscr{S}$ je za týmto účelom zjavne postačujúce vedieť transformovať vstupy $\langle A \rangle, w$ na kód $\langle M_{A,w} \rangle$ nejakého deterministického Turingovho stroja $M_{A,w}$ tak, aby platilo

$$L(M_{A,w}) \in \mathscr{S} \text{ práve vtedy, keď } w \notin L(A). \tag{6}$$

Stroj akceptujúci L_U^C potom jednoducho (za predpokladu, že dostane platný vstup univerzálneho problému) transformuje vstupy $\langle A \rangle, w$ na $\langle M_{A,w} \rangle$ a na tomto kóde spustí stroj pre vlastnosť $\mathscr{S}$, pričom akceptuje práve vtedy, keď akceptuje tento stroj. Základná schéma redukcie (rovnaká ako pre bod (i)) je na obrázku 11.

Zostáva opísať konštrukciu stroja $M_{A,w}$. Keďže nie je splnená podmienka (ii), nutne existuje nekonečný jazyk $L \in \mathscr{S}$ taký, že žiaden konečný podjazyk L_0 jazyka L nie je v $\mathscr{S}$. Nech A_L je deterministický Turingov stroj akceptujúci L, ktorý na vstupe dĺžky n urobí pred akceptáciou aspoň n krokov výpočtu (zrejme ide o normálny tvar).

Stroj $M_{A,w}$ na každom vstupe x najprv odsimuluje výpočet stroja A_L, pričom vo výpočte pokračuje iba ak stroj A_L akceptuje. V takom prípade sa spustí simulácia $|x|$

krokov výpočtu stroja A na vstupe w. Ak stroj A počas simulovaných $|x|$ krokov výpočtu slovo w neakceptuje, stroj $M_{A,w}$ akceptuje slovo x.

Ak teda $w \notin L(A)$, stroj $M_{A,w}$ akceptuje práve všetky slová z jazyka L – slová mimo L sa „odfiltrujú“ simuláciou stroja A_L; slová x z L sa ale akceptujú, pretože stroj A slovo w neakceptuje na žiaden počet krokov, a teda ani na $|x|$ krokov. Ak naopak $w \in L(A)$, stroj $M_{A,w}$ akceptuje iba také slová x z jazyka L, že stroj A slovo w na $|x|$ krokov ešte neakceptuje. Takýchto slov je ale vďaka normálnemu tvaru stroja A_L iba konečne veľa. Platí teda

$$L(M_{A,w}) = \begin{cases} L & \text{ak } w \notin L(A) \\ \text{nejaký konečný podjazyk jazyka } L & \text{inak} \end{cases},$$

čo je silnejšie, než (6). Schéma konštrukcie stroja $M_{A,w}$ je na obrázku 12.

(*iii*) Priamo. Označme symbolom $\mathscr{M}$ množinu všetkých konečných jazykov v $\mathscr{S}$ (nad abecedou $\Sigma = \{0, 1\}$). Ukážeme, že ak je vlastnosť $\mathscr{S}$ rekurzívne vyčísliteľná, tak existuje generujúci Turingov stroj M pre jazyk $L_{\mathscr{M}}$ (definícia 4).

Čitateľ by iste ľahko dokázal zostrojiť generujúci Turingov stroj M' pre jazyk zodpovedajúci množine všetkých konečných jazykov (nad Σ). Stroj M bude pracovať nasledovne:

1. Opakuj v nekonečnom cykle pre $k = 1, 2, \ldots$:
   1.1 Pomocou M' vygeneruj kódy *prvých* k konečných jazykov $L_1, \ldots, L_k$ (nad abecedou Σ).
   1.2 Skonštruuj kódy $\langle A_1 \rangle, \ldots, \langle A_k \rangle$ Turingových strojov $A_1, \ldots, A_k$ takých, že

   $$L(A_1) = L_1, \ldots, L(A_k) = L_k$$

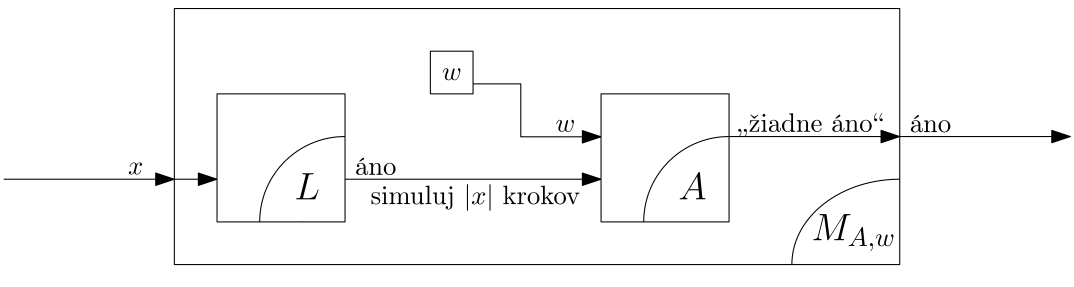
   (detaily tohto kroku prenechávame čitateľovi ako jednoduché cvičenie).
   1.3 Odsimuluj *prvých* k krokov (hypotetického) stroja pre $\mathscr{S}$ postupne na vstupoch $\langle A_1 \rangle, \ldots, \langle A_k \rangle$. Pre každý kód $\langle A_i \rangle$, ktorý stroj pre $\mathscr{S}$ akceptoval, vygeneruj na výstupe kód konečného jazyka L_i.

Je zrejmé, že takýto stroj vygeneruje kód každého konečného jazyka s vlastnosťou $\mathscr{S}$ nekonečne veľa ráz. To však nie je na závadu, keďže definícia generujúceho Turingovho stroja takéto „správanie“ umožňuje.

$\Leftarrow$: Zostáva ukázať, že ak sú splnené podmienky (*i*), (*ii*) a (*iii*), tak je vlastnosť $\mathscr{S}$ rekurzívne vyčísliteľná.

Treba teda skonštruovať Turingov stroj, ktorý akceptuje kód $\langle A \rangle$ deterministického Turingovho stroja A práve vtedy, keď $L(A) \in \mathscr{S}$. Keďže ide iba o akceptáciu (teda o rekurzívnu vyčísliteľnosť), stačí ukázať existenciu *nedeterministického* Turingovho stroja.

Ten môže pracovať napríklad tak, že pre daný kód $\langle A \rangle$ „uhádne“ kód konečného jazyka L_0. Následne odsimuluje stroj A postupne na všetkých slovách z jazyka L_0. Ak stroj A zakaždým



Obr. 12: Schematické znázornenie konštrukcie stroja $M_{A,w}$.

akceptoval, platí $L_0 \subseteq L(A)$. V takom prípade ešte stroj pre $\mathscr{S}$ odsimuluje stroj generujúci kódy všetkých konečných jazykov z $\mathscr{S}$, ktorý existuje vďaka (iii). Ak tento stroj vygeneruje kód jazyka L_0, stroj pre $\mathscr{S}$ akceptuje.

Ľahko vidieť, že takýto stroj skutočne akceptuje všetky kódy $\langle A \rangle$ také, že $L(A) \in \mathscr{S}$. Ak totiž $L(A) \in \mathscr{S}$, tak podľa (ii) musí existovať konečný podjazyk L_0 jazyka $L(A)$ taký, že $L_0 \in \mathscr{S}$. V takom prípade nedeterministický stroj pre $\mathscr{S}$ v aspoň jednej vetve akceptuje. Naopak, ak stroj pre $\mathscr{S}$ akceptuje, tak musel nájsť konečný jazyk L_0 taký, že $L_0 \subseteq L(A)$ a $L_0 \in \mathscr{S}$. Z platnosti podmienky (i) potom vyplýva, že aj $L(A) \in \mathscr{S}$.

Čitateľovi odporúčame zamyslieť sa nad tým, ako by vyzerala konštrukcia *deterministického* Turingovho stroja pre vlastnosť $\mathscr{S}$. $\square$

Použitie druhej Riceovej vety

Príklad 6. Pomocou druhej Riceovej vety možno dokázať napríklad nasledujúce tvrdenia:

- Prázdnosť rekurzívne vyčísliteľných jazykov nie je rekurzívne vyčísliteľná. Jazyk $\emptyset$ je totiž prázdny, ale existuje jazyk $L \supseteq \emptyset$, ktorý prázdny nie je. Podmienka (i) druhej Riceovej vety tak nie je splnená.

- Rovnosť so Σ^* nie je pre rekurzívne vyčísliteľné jazyky rekurzívne vyčísliteľná. Jazyk Σ^* je totiž zjavne rovný Σ^*, no neexistuje žiaden konečný podjazyk Σ^*, ktorý by bol rovný Σ^*. Nie je preto splnená podmienka (ii) druhej Riceovej vety.

- Nech $\mathscr{S}$ je vlastnosť, ktorú majú všetky jazyky L také, že $L - L_U \neq \emptyset$. Množina konečných jazykov s touto vlastnosťou nie je rekurzívne vyčísliteľná, pretože inak by bola rekurzívne vyčísliteľná aj množina všetkých jednoprvkových jazykov $\{w\}$ takých, že $\{w\} - L_U \neq \emptyset$. V dôsledku toho by by bol rekurzívne vyčísliteľný jazyk všetkých slov w takých, že $w \notin L_U$ – a teda by platilo $L_U^C \in \mathscr{L}_{RE}$. Nie je teda splnená podmienka (iii) druhej Riceovej vety a vlastnosť $\mathscr{S}$ nie je rekurzívne vyčísliteľná.

- Neprázdnosť rekurzívne vyčísliteľných jazykov je rekurzívne vyčísliteľná, čo možno dokázať napríklad overením podmienok (i) až (iii) druhej Riceovej vety.