Meze rozhodnutelnosti

Existují jazyky (problémy), jež nejsou rekurzivně vyčíslitelné (částečně rozhodnutelné)?

Které jazyky, resp. problémy, nejsou rekurzívní (rozhodnutelné)?

Jazyky mimo třídu 0

Existence jazyků mimo třídu 0

Věta 9.1 Pro každou abecedu Σ existuje jazyk nad Σ, který není typu 0 (tj. rekurzívně vyčíslitelný).

Důkaz.

1. Libovolný jazyk typu 0 nad Σ může být přijat TS s $\Gamma = \Sigma \cup \{\Delta\}$: Pokud M používá více symbolů, můžeme je zakódovat jako jisté posloupnosti symbolů ze $\Sigma \cup \{\Delta\}$ a sestrojit TS M', který simuluje M.
2. Nyní můžeme snadno systematicky vypisovat všechny TS s $\Gamma = \Sigma \cup \{\Delta\}$. Začneme stroji se dvěma stavy, pak se třemi stavy, ...
   Závěr: Množina všech takových strojů a tedy i jazyků typu 0 je spočetná.
3. Množina Σ^* ale obsahuje nekonečně mnoho řetězců a proto je množina 2^{Σ^*} zahrnující všechny jazyky nespočetná – důkaz viz další strana.
4. Z rozdílnosti mohutností spočetných a nespočetných množin plyne platnost uvedené věty.

$\Box$

Lemma 9.1 Pro neprázdné, konečné Σ je množina 2^{Σ^*} nespočetná.

Důkaz. Důkaz provedeme tzv. diagonalizací (poprvé použitou Cantorem při důkazu rozdílné mohutnosti $\mathbb{N}$ a $\mathbb{R}$).

- Předpokládejme, že 2^{Σ^*} je spočetná. Pak dle definice spočetnosti existuje bijekce $f : \mathbb{N} \longleftrightarrow 2^{\Sigma^*}$.
- Uspořádejme Σ^* do nějaké posloupnosti $w_1, w_2, w_3, \ldots$, např. ε, x, y, xx, xy, yx, yy, xxx, ... pro $\Sigma = \{x, y\}$. Nyní můžeme f zobrazit nekonečnou maticí:

$$\begin{array}{lcccccc}
 & w_0 & w_1 & w_2 & \ldots & w_i & \ldots \\
L_0 = f(0) & a_{00} & a_{01} & a_{02} & \ldots & a_{0i} & \ldots \\
L_1 = f(1) & a_{10} & a_{11} & a_{12} & \ldots & a_{1i} & \ldots \\
L_2 = f(2) & a_{20} & a_{21} & a_{22} & \ldots & a_{2i} & \ldots \\
\ldots & & & & & &
\end{array}
\text{, kde } a_{ij} = \begin{cases} 0, \text{ jestliže } w_j \notin L_i, \\ 1, \text{ jestliže } w_j \in L_i. \end{cases}$$

- Uvažujme jazyk $\overline{L} = \{w_i \mid a_{ii} = 0\}$. $\overline{L}$ se liší od každého jazyka $L_i = f(i)$, $i \in \mathbb{N}$:
  - je-li $a_{ii} = 0$, pak w_i patří do jazyka,
  - je-li $a_{ii} = 1$, pak w_i nepatří do jazyka.
- Současně ale $\overline{L} \in 2^{\Sigma^*}$, f tudíž není surjektivní, což je spor.

$\square$

Problém zastavení

Kódování TS

❖ Kódovací systém pro TS zahrnuje (1) kódování stavů (tak, aby byly odlišeny všechny stavy včetně q_0 a q_F), (2) symbolů z Γ a (3) přechodové funkce δ.

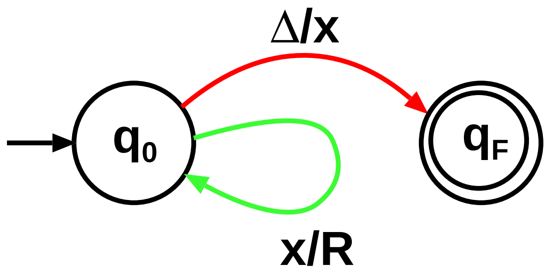
❖ Kódování stavů: Množinu stavů Q uspořádáme do posloupnosti q_0, q_F, q, p, ..., t. Stav q_j zakódujeme jako 0^j, přičemž indexujeme (např.) od nuly.

❖ Kódování symbolů a příkazů L/R: Předpokládejme, že $\Gamma = \Sigma \cup \{\Delta\}$. Uspořádáme Σ do posloupnosti a_1, a_2, ..., a_n a zvolíme tyto kódy: $\Delta \mapsto \varepsilon$, $L \mapsto 0$, $R \mapsto 00$, $a_i \mapsto 0^{i+2}$.

❖ Přechod $\delta(p, x) = (q, y)$, kde $y \in \Gamma \cup \{L, R\}$, reprezentujeme čtveřicí (p, x, q, y) a kódujeme zřetězením kódů p, x, q, y při použití 1 jako oddělovače, tj. jako $\langle p \rangle 1 \langle x \rangle 1 \langle q \rangle 1 \langle y \rangle$, kde $\langle \_ \rangle$ značí kód _.

❖ Celý TS kódujeme jako posloupnost kódů přechodů oddělených a ohraničených 1.

Příklad 9.1



kód: $1\,1101000\,1\,10001100\,1$

Univerzální TS

❖ Zavádí koncept „programovatelného“ stroje, který umožňuje ve vstupním řetězci specifikovat konkrétní TS (tj. program) i data, nad nimiž má tento stroj pracovat.

❖ TS, který má být simulován, budeme kódovat, jak bylo uvedeno na předchozí straně, vstupní řetězec budeme kódovat jako posloupnost příslušných kódů symbolů oddělených a ohraničených 1. Kód stroje a vstupního řetězce oddělíme např. $\#$.

Příklad 9.2 TS z předchozí strany mající na vstupu xxx:

$$1110100011000110011\#1000100010001$$

❖ Univerzální TS, který zpracuje toto zadání můžeme navrhnout jako třípáskový stroj, který

- má na 1. pásce zadání (a později výstup),
- 2. pásku používá k simulaci pracovní pásky původního stroje a
- na 3. pásce má zaznamenán řídicí stav simulovaného stroje a aktuální pozici hlavy (pozice hlavy i je kódována jako 0^i).

❖ Univerzální stroj pracuje takto:

1. Stroj zkontroluje, zda vstup odpovídá nějakému $M\#w$ a pokud ne, abnormálně zastaví.
2. Přepíše w na 2. pásku, na 3. pásku umístí kód q_0 a za něj poznačí, že hlava se nachází na levém okraji pásky.
3. Na 2. pásce vyhledá aktuální symbol pod hlavou simulovaného stroje a na 1. pásce vyhledá přechod proveditelný ze stavu zapsaného na začátku 3. pásky pro tento vstupní symbol. Pokud žádný přechod možný není, stroj abnormálně zastaví.
4. Stroj provede na 2. a 3. pásce změny odpovídající simulovanému přechodu (přepis aktuálního symbolu, změna pozice hlavy, změna řídicího stavu).
5. Pokud nebyl dosažen stav q_F simulovaného stroje, přejdeme na bod 3. Jinak stroj vymaže 1. pásku, umístí na ní obsah 2. pásky a zastaví přechodem do svého koncového stavu.

❖ Víme, že výše uvedený stroj můžeme převést na jednopáskový univerzální TS, který budeme v dalším značit jako T_U.

Problém zastavení TS

Věta 9.2 **Problém zastavení TS (Halting Problem)**, kdy nás zajímá, zda daný TS M pro danou vstupní větu w zastaví, **není rozhodnutelný**, ale je **částečně rozhodnutelný**.

Důkaz.

- Problému zastavení odpovídá rozhodování jazyka $HP = \{\langle M \rangle \# \langle w \rangle \mid M \text{ zastaví při } w\}$, kde $\langle M \rangle$ je kód TS M a $\langle w \rangle$ je kód w.
- Částečnou rozhodnutelnost ukážeme snadno použitím modifikovaného T_U, který zastaví přijetím vstupu $\langle M \rangle \# \langle w \rangle$ právě tehdy, když M zastaví při w – modifikace spočívá v převedení abnormálního zastavení při simulaci na zastavení přechodem do q_F.
- Nerozhodnutelnost ukážeme pomocí diagonalizace:
  1. Pro $x \in \{0, 1\}^*$, nechť M_x je TS s kódem x, je-li x legální kód TS. Jinak ztotožníme M_x s pevně zvoleným TS, např. TS, který pro libovolný vstup okamžitě zastaví.
  2. Můžeme nyní sestavit posloupnost $M_\varepsilon, M_0, M_1, M_{00}, M_{01}, M_{10}, M_{11}, M_{000}, \ldots$ zahrnující všechny TS nad $\Sigma = \{0, 1\}$ indexované řetězci z $\{0, 1\}^*$.

Důkaz pokračuje dále.

Pokračování důkazu.

3. Uvažme nekonečnou matici

	ε	0	1	00	01	10	...
M_ε	$H_{M_\varepsilon,\varepsilon}$	$H_{M_\varepsilon,0}$	$H_{M_\varepsilon,1}$	$H_{M_\varepsilon,00}$	$H_{M_\varepsilon,01}$	...	
M_0	$H_{M_0,\varepsilon}$	$H_{M_0,0}$	$H_{M_0,1}$	$H_{M_0,00}$	$H_{M_0,01}$	...	
M_1	$H_{M_1,\varepsilon}$	$H_{M_1,0}$	$H_{M_1,1}$	$H_{M_1,00}$	$H_{M_1,01}$	...	
M_{00}	$H_{M_{00},\varepsilon}$	$H_{M_{00},0}$	$H_{M_{00},1}$	$H_{M_{00},00}$	$H_{M_{00},01}$	...	
M_{01}	$H_{M_{01},\varepsilon}$	$H_{M_{01},0}$	$H_{M_{01},1}$	$H_{M_{01},00}$	$H_{M_{01},01}$	...	
...							

kde $H_{M_x,y} = \begin{cases} \textbf{C}\text{, jestliže } M_x \text{ cyklí na } y, \\ \textbf{Z}\text{, jestliže } M_x \text{ zastaví na } y. \end{cases}$

4. Předpokládejme, že existuje *úplný* TS K přijímající jazyk HP, tj. K pro vstup $\langle M\rangle\#\langle w\rangle$
   - zastaví normálně (přijme) právě tehdy, když M zastaví na w,
   - zastaví abnormálně (odmítne) právě tehdy, když M cyklí na w.
5. Sestavíme TS N, který pro vstup $x \in \{0,1\}^*$:
   - Sestaví M_x z x a zapíše $\langle M_x\rangle\#x$ na svou pásku.
   - Simuluje K na $\langle M_x\rangle\#x$, přijme, pokud K odmítne, a přejde do nekonečného cyklu, pokud K přijme.

Důkaz pokračuje dále.

Pokračování důkazu.

Všimněme si, že N v podstatě komplementuje diagonálu uvedené matice:

$$
\begin{array}{lllllll}
 & \varepsilon & 0 & 1 & 00 & 01 & 10 \quad \ldots \\
M_\varepsilon & H_{M_\varepsilon,\varepsilon} & H_{M_\varepsilon,0} & H_{M_\varepsilon,1} & H_{M_\varepsilon,00} & H_{M_\varepsilon,01} & \ldots \\
M_0 & H_{M_0,\varepsilon} & H_{M_0,0} & H_{M_0,1} & H_{M_0,00} & H_{M_0,01} & \ldots \\
M_1 & H_{M_1,\varepsilon} & H_{M_1,0} & H_{M_1,1} & H_{M_1,00} & H_{M_1,01} & \ldots \\
M_{00} & H_{M_{00},\varepsilon} & H_{M_{00},0} & H_{M_{00},1} & H_{M_{00},00} & H_{M_{00},01} & \ldots \\
M_{01} & H_{M_{01},\varepsilon} & H_{M_{01},0} & H_{M_{01},1} & H_{M_{01},00} & H_{M_{01},01} & \ldots \\
\ldots & & & & & &
\end{array}
$$

6. Dostáváme, že

$$
\begin{array}{llll}
N \text{ zastaví na } x & \Leftrightarrow & K \text{ odmítne } \langle M_x \rangle \# \langle x \rangle & \text{(definice } N\text{)} \\
 & \Leftrightarrow & M_x \text{ cyklí na } x & \text{(předpoklad o } K\text{).}
\end{array}
$$

7. To ale znamená, že N se liší od každého M_x alespoň na jednom řetězci – konkrétně x. Což je ovšem spor s tím, že posloupnost $M_\varepsilon, M_0, M_1, M_{00}, M_{01}, M_{10}, M_{11}, M_{000}, \ldots$ zahrnuje všechny TS nad $\Sigma = \{0, 1\}$. Tento spor plyne z předpokladu, že existuje TS K, který pro daný TS M a daný vstup x určí (rozhodne), zda M zastaví na x, či nikoliv.

$\square$

❖ Ukázali jsme, že problém zastavení TS je částečně rozhodnutelný a tedy jazyk HP rekurzívně vyčíslitelný. Z věty 8.6 pak plyne, že **komplement problému zastavení** není ani částečně rozhodnutelný a jazyk co-$HP = \{\langle M\rangle\#\langle w\rangle \mid M$ nezastaví při $w\}$ je příkladem jazyka, jenž není ani rekurzívně vyčíslitelný.

S dalším příkladem takového jazyka se seznámíme v následujícím problému.

Redukce

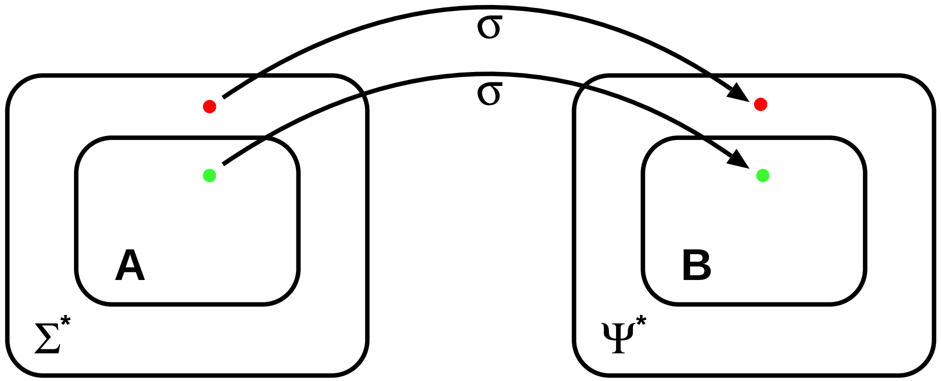
Důkaz nerozhodnutelnosti redukcí

❖ Technika redukce patří spolu s diagonalizací k nejpoužívanějším technikám důkazu, že nějaký problém není rozhodnutelný (částečně rozhodnutelný) – neboli, že určitý jazyk není rekurzívní (rekurzívně vyčíslitelný):

- víme, že jazyk A není rekurzívní (rekurzívně vyčíslitelný),
- zkoumáme jazyk B,
- ukážeme, že A lze úplným TS převést (redukovat) na B,
- to ale znamená, že B rovněž není rekurzívní (rekurzívně vyčíslitelný) – jinak by šlo použít úplný TS (ne-úplný TS) přijímající B a příslušné redukce k sestavení úplného TS (ne-úplného TS) přijímajícího A, což by byl spor.

❖ Argumentace výše samozřejmě ukazuje, že redukci lze použít i při dokazování, že určitý problém je rekurzívní (částečně rekurzívní).

Definice 9.1 Nechť A, B jsou jazyky, $A \subseteq \Sigma^*$, $B \subseteq \Psi^*$. Redukce jazyka A na jazyk B je totální, rekurzívně vyčíslitelná funkce $\sigma : \Sigma^* \to \Psi^*$ taková, že $\forall w \in \Sigma^*. w \in A \Leftrightarrow \sigma(w) \in B$.



❖ Existuje-li redukce jazyka A na B, říkáme, že A je redukovatelný na B, což značíme $A \leq B$.

Věta 9.3 Nechť $A \leq B$.

1. Není-li jazyk A rekurzívně vyčíslitelný, pak ani jazyk B není rekurzívně vyčíslitelný.
2. Není-li jazyk A rekurzívní, pak ani jazyk B není rekurzívní.

$\overline{1}$. Je-li jazyk B rekurzívně vyčíslitelný, pak i jazyk A je rekurzívně vyčíslitelný.

$\overline{2}$. Je-li jazyk B rekurzívní, pak i jazyk A je rekurzívní.

Důkaz. Dokážeme, že pokud $A \leq B$, pak ($\overline{1}$) je-li jazyk B rekurzívně vyčíslitelný, pak i jazyk A je rekurzívně vyčíslitelný:

- Nechť M_R je úplný TS počítající redukci σ z A na B a M_B je TS přijímající B.
- Sestrojíme M_A přijímající A:
  1. M_A simuluje M_R na vstupu w, což transformuje obsah pásky na $\sigma(w)$.
  2. M_A simuluje výpočet M_B na $\sigma(w)$.
  3. Pokud M_B zastaví a přijme, M_A rovněž zastaví a přijme, jinak M_A zastaví abnormálně nebo cyklí.
- Zřejmě platí:

$$\begin{array}{lcll} M_A \text{ přijme } w & \Leftrightarrow & M_B \text{ přijme } \sigma(w) & \\ & \Leftrightarrow & \sigma(w) \in B & \\ & \Leftrightarrow & w \in A & \qquad\qquad \text{(definice redukce).} \end{array}$$

Tvrzení (1) je kontrapozicí ($\overline{1}$); tvrzení ($\overline{2}$) dokážeme podobně jako ($\overline{1}$) při použití úplného TS M_B; tvrzení (2) je kontrapozicí ($\overline{2}$).

Kontrapozice: $p \rightarrow q \Leftrightarrow \neg q \rightarrow \neg p$ (Modus tollens)

$\square$

Problém náležitosti a další problémy

Problém náležitosti pro $\mathcal{L}_0$

Věta 9.4 **Problém náležitosti (Membership Problem)** řetězce w do jazyka L typu 0 **není rozhodnutelný**, ale **je částečně rozhodnutelný**.

Důkaz. Částečná rozhodnutelnost je zřejmá: jednoduše použijeme T_U, který bude simulovat TS M, $L(M) = L$, nad daným řetězcem w. Nerozhodnutelnost ukážeme redukcí z problému zastavení:

- Libovolný TS M můžeme snadno upravit na M', který přijme w právě tehdy, když M zastaví na vstup w přijetím nebo odmítnutím w: postačí dodat veškeré „chybějící" přechody jejich svedením do q_F, dodat počáteční poznačení levého okraje unikátním symbolem a přechod do q_F kdykoliv se hlava posune na tento symbol.
- Uvedenou transformaci lze zřejmě snadno realizovat na úrovni kódů Turingových strojů *úplným TS* – ten tak bude implementovat redukci jazyka HP na jazyk $MP = \{\langle M''\rangle\#\langle w''\rangle \mid w'' \in L(M'')\}$.
- Jsme tedy schopni redukovat HP úplným TS na MP a současně víme, že HP není rekurzívní. Dle věty 10.3 (2) tedy máme, že MP není rekurzívní a tedy problém náležitosti pro jazyky typu 0 není rozhodnutelný.

□

- Podobně jako u problému zastavení nyní z věty 8.6 plyne, že
  - **komplement problému náležitosti** není ani částečně rozhodnutelný a
  - jazyk co-$MP = \{\langle M'' \rangle \# \langle w'' \rangle \mid w'' \notin L(M'')\}$ je dalším příkladem jazyka, jenž není ani rekurzívně vyčíslitelný.

Příklady dalších problémů pro TS

❖ Konstrukcí příslušného úplného TS (a v případě složitější konstrukce důkazem její korektnosti) lze ukázat, že např. následující problémy jsou rozhodnutelné:

- Daný TS má alespoň 2005 stavů.
- Daný TS učiní více než 2005 kroků na vstupu ε.
- Daný TS učiní více než 2005 kroků na *nějakém* vstupu.

❖ Konstrukcí příslušného (ne-úplného) TS a důkazem nerekurzívnosti redukcí lze ukázat, že např. následující problémy jsou částečně rozhodnutelné:

- Jazyk daného TS je neprázdný.
- Jazyk daného TS obsahuje alespoň 2005 slov.

❖ Důkazem redukcí, že jazyky odpovídající následujícím problémům nejsou ani parciálně rekurzívní lze ukázat, že např. následující problémy nejsou ani částečně rozhodnutelné:

- Jazyk daného TS je prázdný.
- Jazyk daného TS obsahuje nanejvýš 2005 slov.
- Jazyk daného TS je konečný (regulární, bezkontextový, kontextový, rekurzívní).