

Základy složitosti a vyčíslitelnosti

NTIN090

Petr Kučera

2024/25 (3. přednáška)

Dnešní plán

- Univerzální Turingův stroj
- Nerozhodnutelnost jazyka univerzálního Turingova stroje
- Problém zastavení
- Převoditelnost
- Důkaz Riceovy věty

Kódování Turingových strojů (připomenutí)

Pár technických omezení

Omezíme se na Turingovy stroje, které

- 1 mají **jediný přijímající stav** a
- 2 mají pouze **binární vstupní abecedu** $\Sigma_{in} = \{0, 1\}$.

- Vstupní řetězce budou zapsány jen pomocí znaků 0 a 1
- Pracovní abecedu nijak neomezujeme
 - Turingův stroj může během výpočtu zapisovat na pásku libovolné symboly
- Jakoukoli konečnou abecedu lze zakódovat do binární abecedy
- Každý TS M lze upravit tak, aby splňoval obě omezení

Zakódování přechodové funkce

$$M = (Q, \Sigma, \delta, q_0, F = \{q_1\})$$



δ zapíšeme řetězcem v_M v abecedě
 $\Gamma = \{0, 1, L, N, R, |, \#, ;\}$



v_M

Každý znak v_M zapíšeme pomocí 3 bitů



$\langle M \rangle$

Binární řetězec reprezentující M

Zápis přechodové funkce v abecedě Γ

- Předpoklady

- $Q = \{q_0, q_1, \dots, q_r\}$ pro nějaké $r \geq 1$, kde q_0 je počáteční stav a q_1 je jediný přijímající stav.
- $\Sigma = \{X_0, X_1, X_2, \dots, X_s\}$ pro nějaké $s \geq 2$, kde $X_0 = 0$, $X_1 = 1$ a $X_2 = \lambda$

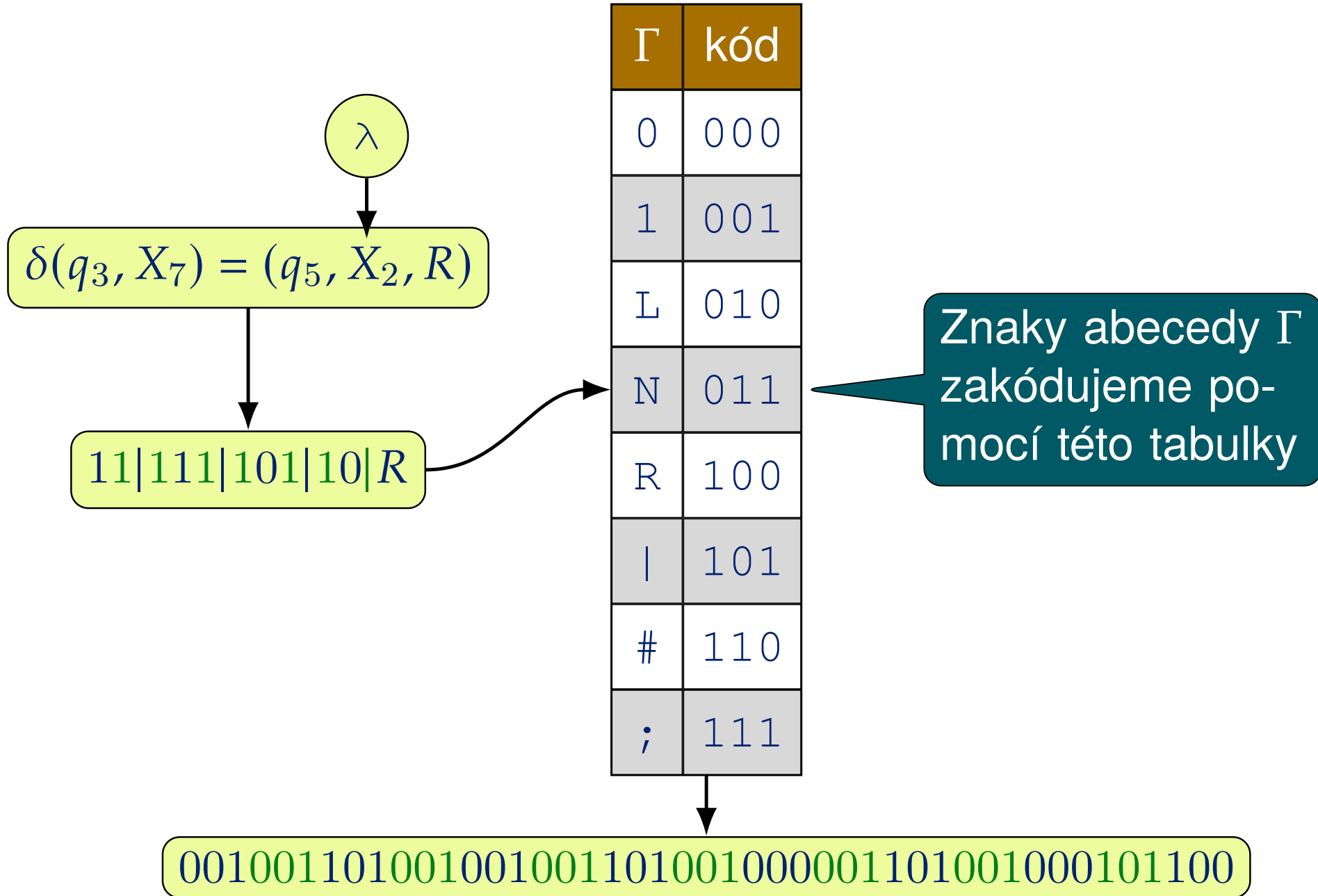
- Instrukci $\delta(q_i, X_j) = (q_k, X_l, Z)$, kde $Z \in \{L, N, R\}$ zakódujeme řetězcem

$$(i)_B|(j)_B|(k)_B|(l)_B|Z$$

- Jsou-li $C_1, \dots, C_n$ kódy instrukcí TS M , pak přechodovou funkci δ zakódujeme řetězcem

$$C_1\#C_2\#\dots\#C_n$$

Převod do binární abecedy



Gödelovo číslo

$\langle M \rangle$ binární řetězec kódující TS M

Gödelovo číslo jednoznačně přiřazené danému Turingovu stroji

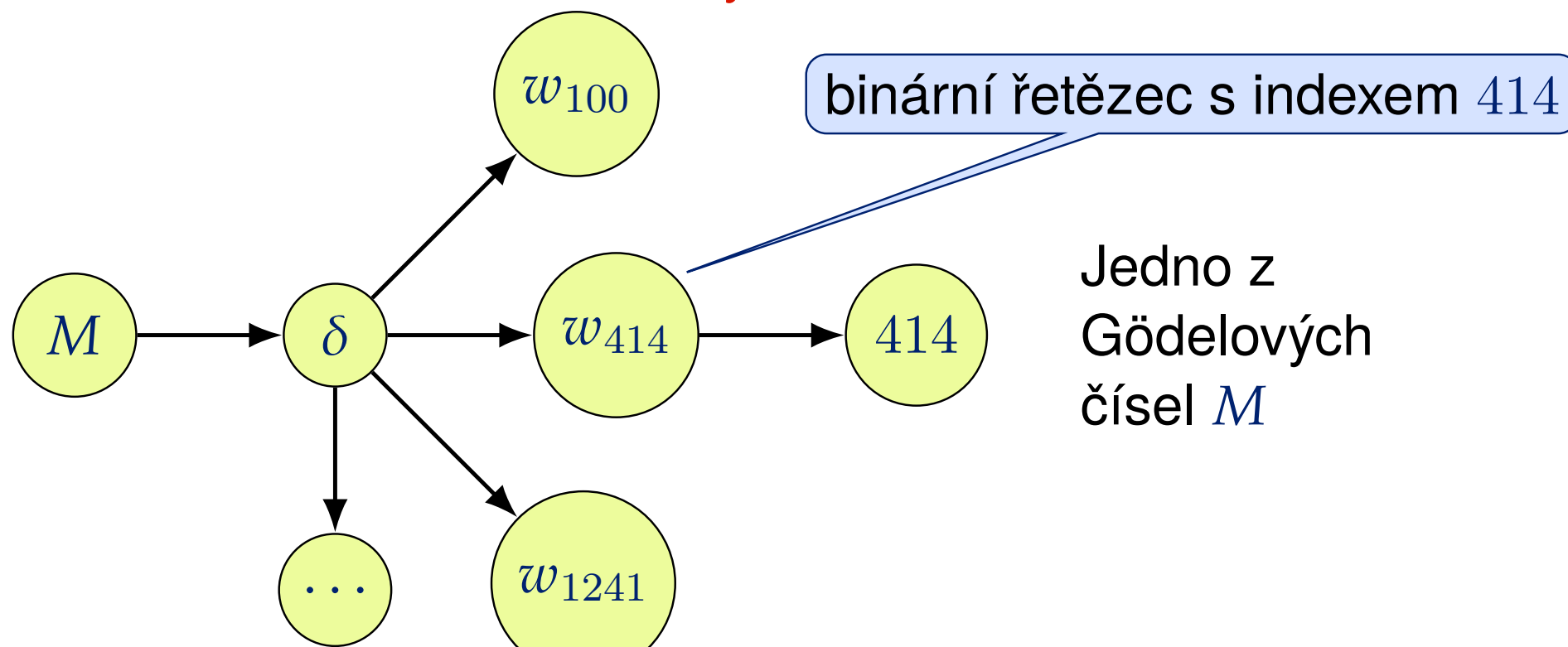
- Definujeme jako $\text{index}(\langle M \rangle)$

Je-li w řetězec, který není syntakticky správným kódem Turingova stroje, přiřadíme mu Turingův stroj M s **prázdnou přechodovou funkcí**.

- přechodová funkce M není definovaná pro žádný vstup a
- M okamžitě odmítne každý vstup, tedy $L(M) = \emptyset$

Nejednoznačnost kódu TS

- Kód TS není jednoznačný, protože nezáleží na
 - pořadí instrukcí,
 - na očíslování stavů kromě počátečního a přijímajícího,
 - znaků páskové abecedy kromě 0 , 1 , λ , a
 - binární zápis čísla stavu nebo znaku může být uvozen libovolným počtem 0 .
- Každý TS má **nekonečně mnoho různých kódů** a potažmo **nekonečně mnoho Gödelových čísel**.



Kódování objektů (značení)

- Konečné objekty (např. číslo, řetězec, Turingův stroj, RAM, graf nebo formuli) můžeme kódovat binárními řetězci
- Podobně můžeme zakódovat i n -tice objektů

Definice

$\langle X \rangle$ binární řetězec kódující objekt X

$\langle X_1, \dots, X_n \rangle$ binární řetězec kódující n -tici objektů $X_1, \dots, X_n$

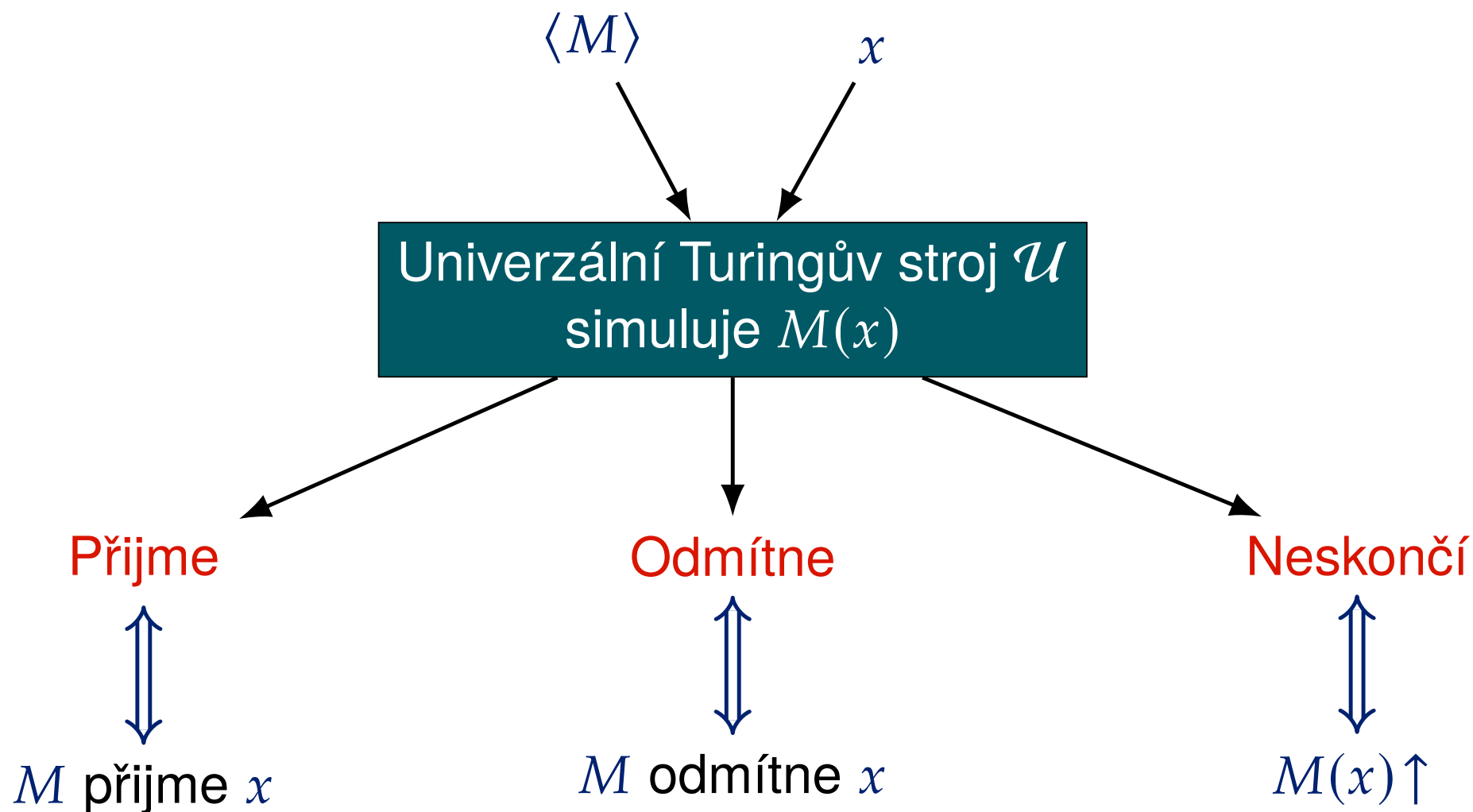
Příklad

$\langle M \rangle$ kód Turingova stroje M

$\langle M, x \rangle$ kód dvojice tvořené Turingovým strojem M a řetězcem x

Univerzální Turingův stroj

Univerzální Turingův stroj



Univerzální Turingův stroj

Vstup $\langle M, x \rangle$ (M je Turingův stroj, x je vstup)

Univerzální Turingův stroj simuluje práci stroje M nad vstupem x

Výsledek práce zastavení/přijetí/zamítnutí vstupu a obsah výstupní pásky je dán výsledkem $M(x)$

PŘIJETÍ VSTUPU

Instance: Kód Turingova stroje M a vstupní řetězec x

Otázka: Přijme M vstup x ?

Univerzální jazyk

Jazyk univerzálního Turingova stroje

$$L_u = \{ \langle M, x \rangle \mid x \in L(M) \}$$

Popíšeme 3-páskový Univerzální Turingův stroj $\mathcal{U}$

1. páska obsahuje vstup $\langle M, x \rangle$

$\langle M \rangle$	x
---------------------	-----

$\hookrightarrow$ tedy kódový Turingův stroj

$\hookrightarrow$ tedy máme jeho vstupní slovo, pro které chceme

Na 2. pásce je uložen obsah pracovní pásky M spustit výpočet

Symbol X_j zapsán jako $(j)_B$, bloky mají touž délku b bitů

...		0	1	0		0	1	0		1	0	1		1	1	0		...
-----	--	---	---	---	--	---	---	---	--	---	---	---	--	---	---	---	--	-----

3. páska obsahuje $(i)_B$ reprezentující aktuální stav q_i stroje M

...	1	0	0	1	1	λ	λ	λ	λ	λ	λ	λ	λ	λ	λ	λ	...
-----	---	---	---	---	---	-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----

Výpočet $\mathcal{U}$

- Vstup $\mathcal{U}$ má dvě části $\langle M \rangle$ a x
 - $\mathcal{U}$ umí číst každou zvlášť
- Simulovaný TS $M = (Q, \Sigma, \delta, q_0, \{q_1\})$
 - Jediný přijímající stav q_1
 - Vstupní abeceda $\{0, 1\}$
 - Pásková abeceda Σ není omezená
- $\langle M \rangle$ kóduje přechodovou funkci δ
- Výpočet $\mathcal{U}(\langle M \rangle, x)$ má 3 fáze
 - 1 Inicializace
 - 2 Simulace
 - 3 Zakončení

Inicializace

1 Syntaktická kontrola

- Pokud první část vstupu není syntakticky správným kódem Turingova stroje, odmítni

2 Určení délky bloku b pro znak na 2. pásce

- Maximální délka znaku X_i v rámci nějaké instrukce
- Abeceda Σ obsahuje alespoň 0 , 1 a λ , tedy $b \geq 2$
- Pracovní abeceda není jinak omezená

3 Přepis vstupu na 2. pásku

- Překódování vstupu do bloků délky b oddělených $|$
- 0 je přepsáno na 0^b ($X_0 = 0$)
- 1 je přepsáno na $0^{b-1}1$ ($X_1 = 1$)

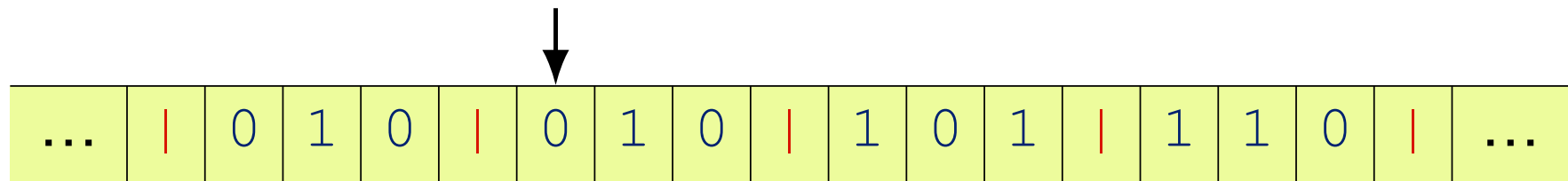
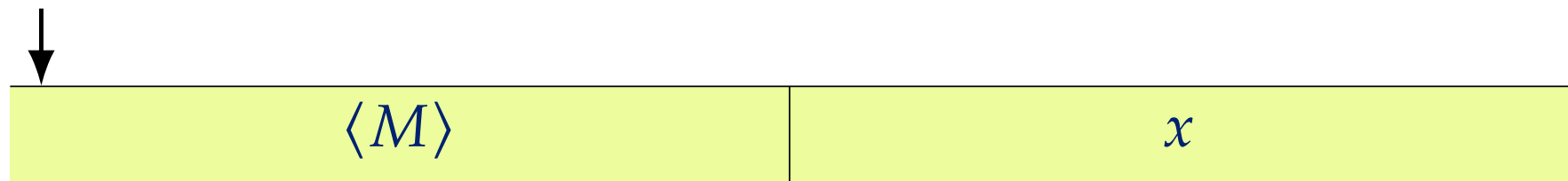
4 Zapiš 0 na 3. pásku

- Počáteční stav je q_0

5 Návrat všech tří hlav na začátky slov na příslušných páskách

Polohy hlav na začátku simulace kroku M

1. páska na začátku kódu $\langle M \rangle$
2. páska nad blokem symbolu X_j , nad nímž je hlava M
3. páska na začátku čísla stavu q_i



$(j)_B$ kóduje symbol X_j pod hlavou M



$(i)_B$ kóduje stav q_i stroje M

Simulace kroku M

- 1 Hledej v $\langle M \rangle$ instrukci pro displej (q_i, X_j)
 - Instrukce není nalezena $\implies$ simulace končí
 - Jinak označme nalezenou instrukci $\delta(q_i, X_j) = (q_k, X_l, Z)$
- 2 3. páska přepiš číslo stavu na $(k)_B$
- 3 2. páska přepiš blok pod hlavou na $(l)_B$ (b bitů)
- 4 2. páska přesuň hlavu
 - o blok vlevo (je-li $Z = L$)
 - o blok vpravo (je-li $Z = R$)
 - na začátek stávajícího bloku (je-li $Z = N$)
- 5 Pokud se hlava přesunem dostala mimo použitou část pásky, $\mathcal{U}$ přidá další blok tvaru $0^{b-2}10$ ($X_2 = \lambda$)
- 6 Vrať hlavy do předpokládaných pozic
- 7 Pokračuj simulací dalšího kroku M

Zakončení

- Obsahuje 3. páska číslo 1?
 - ✓ přijmi (jediný přijímající stav q_1)
 - ✗ odmítni
- Počítá M funkci?
 - Přepiš obsah 2. pásky do řetězce z Σ^*

Nerozhodnutelnost Univerzálního jazyka

Vlastnosti univerzálního jazyka

Věta

Jazyk $L_u = \{\langle M, x \rangle \mid x \in L(M)\}$ je částečně rozhodnutelný, ale není rozhodnutelný.

- Částečná rozhodnutelnost $\iff$ univerzální Turingův stroj
- Nerozhodnutelnost ukážeme **diagonalizací**
 - 1 Univerzální jazyk reprezentujeme jako matici A
 - 2 Jazyk daný doplnkem diagonály A není částečně rozhodnutelný
 - 3 Z toho dovodíme, že L_u není rozhodnutelný

Univerzální jazyk jako matice

$L_u = \{\langle M, x \rangle \mid x \in L(M)\}$ lze reprezentovat nekonečnou maticí A

w_i označuje binární řetězec s indexem i

M_i označuje TS s Gödelovým číslem i

Platí: $w_i = \langle M_i \rangle$

- jeden TS tam má nekonečně mnoho řádků
 - existuje nekonečně mnoho repr. TS
- ale každý TS tu má vlastní řádek

Gödelova čísla

Odpovídá Turingovu stroji M_i s Gödelovým číslem i

Odpovídá j -tému binárnímu řetězci w_j s indexem j

indexy binárních řetězců

A	0	1	2		i		j	
0	0	1	0		0		1	
1	1	1	0		1		0	
2	1	1	0		0		1	
.....								
i	0	0	1		1		0	

$w_i \in L(M_i)$

$w_j \notin L(M_i)$

Matice univerzálního jazyka

- Každý Turingův stroj M má nekonečně mnoho Gödelových čísel
- ⇒ A obsahuje nekonečně mnoho řádků odpovídajících
- konkrétnímu Turingovu stroji M
 - konkrétnímu částečně rozhodnutelnému jazyku

Doplňěk diagonály matice A určuje **diagonální jazyk**

$$\text{DIAG} = \{ \langle M \rangle \mid \langle M \rangle \notin L(M) \}$$

- DIAG nemá svůj řádek v matici A
- DIAG není částečně rozhodnutelný

0	1	0	...	0	...	1	...
1	1	0	...	1	...	0	...
1	1	0	...	0	...	1	...
0	0	1	...	1	...	0	...
...	...	...	...	...	...	...	...
1	0	1	...	0	...	1	...

– je ale důležité, že se liší na nekonečně mnoha pozicích

Od každého řádku se ale musí lišit, protože tvoří doplněk té diagonály. Proto není řádek součástí A .
– jinak bych pro konečně mnoho řádků mohl proměnit jejich přijetí

Diagonální jazyk

$$\text{DIAG} = \{\langle M \rangle \mid \langle M \rangle \notin L(M)\}$$

$$\text{DIAG} = \{w_i \mid w_i \notin L(M_i)\}$$

Gödelova čísla

indexy binárních řetězců

Odpovídá j -tému binárnímu řetězci w_j s indexem j

A	0	1	2	...	i	...	j
0	0	1	0	...	0	...	1
1	1	1	0	...	1	...	0
2	1	1	0	...	0	...	1
...	...	...	...	...	...	...	...
i	0	0	1	...	1	...	0
...	...	...	...	...	...	...	...
DIAG	1	0	1	...	0	...	1

$w_i \in L(M_i)$

Odpovídá Turingovu stroji M_i s Gödelovým číslem i

DIAG se liší od každého řádku matice A v diagonálním políčku

$w_i \notin L(M_i)$

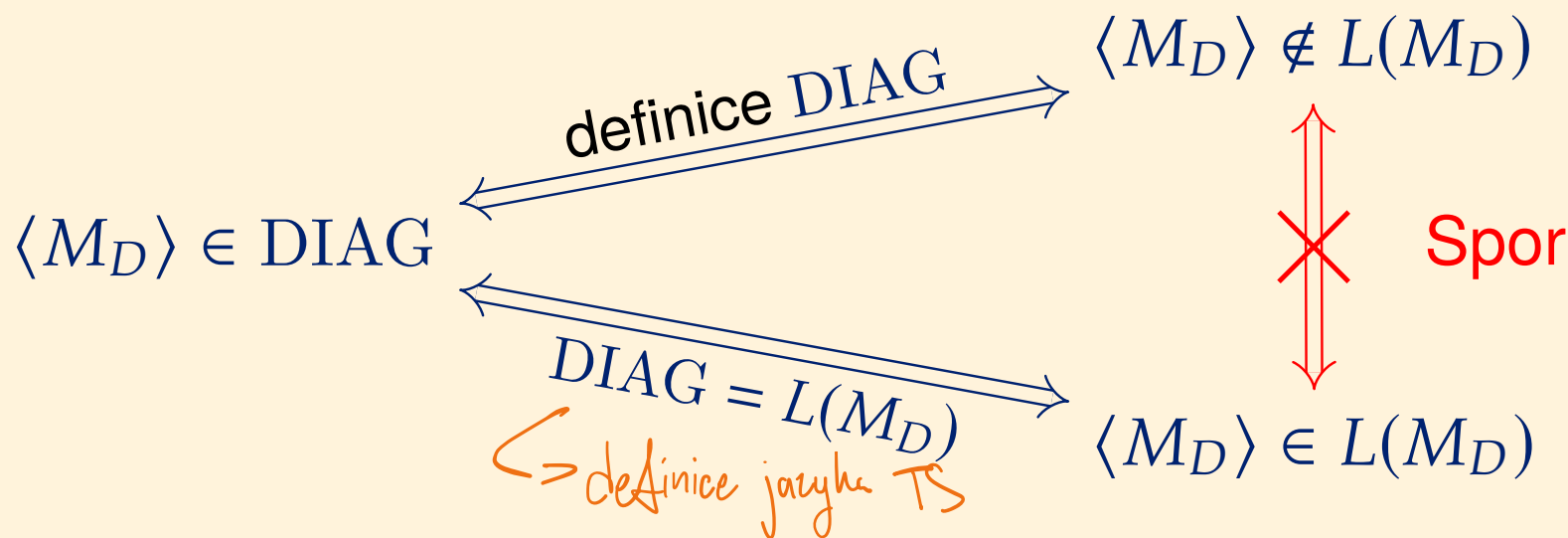
Diagonální jazyk není částečně rozhodnutelný

Věta

Jazyk $\text{DIAG} = \{\langle M \rangle \mid \langle M \rangle \notin L(M)\}$ není částečně rozhodnutelný

Důkaz.

Sporem: existuje TS M_D , který přijímá DIAG (tj. $\text{DIAG} = L(M_D)$)



□

Nerozhodnutelnost univerzálního jazyka

Věta

Jazyk $L_u = \{\langle M, x \rangle \mid x \in L(M)\}$ není rozhodnutelný.

Důkaz.

- **Sporem:** Existuje Turingův stroj M_u , který rozhoduje L_u
 - $L_u = L(M_u)$ a $M_u(\langle M, x \rangle) \downarrow$ pro každý vstup $\langle M, x \rangle$
- Pro každý Turingův stroj M platí $\langle M \rangle \in L(M) \iff \langle M, \langle M \rangle \rangle \in L_u$

$$\begin{array}{ccccc} & \text{definice DIAG} & & \text{definice } L_u & \\ \langle M \rangle \in \text{DIAG} & \iff & \langle M \rangle \notin L(M) & \iff & \langle M, \langle M \rangle \rangle \notin L_u \end{array}$$

- Stroj M_u lze použít k rozhodování DIAG $\longrightarrow$ když M_u přijme, zamítne, když zamítne, přijme.
- **Spor** s nerozhodnutelností DIAG $\searrow \longrightarrow$ Vyřešil bych tím i toto ale...

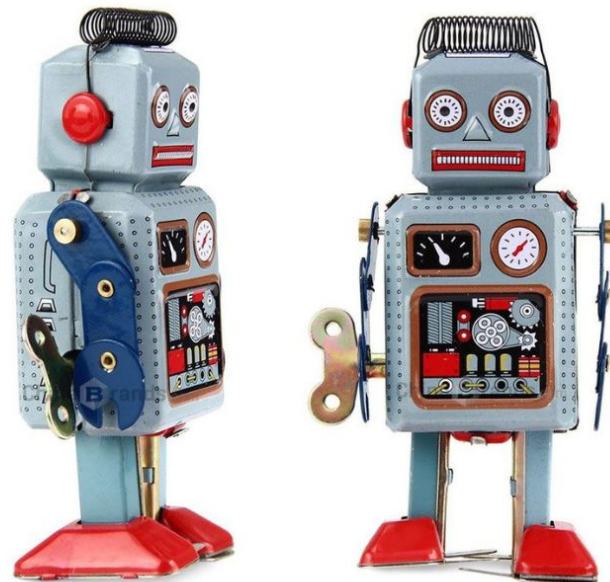


Převoditelnost

Soutěž pro středně pokročilé

Soutěž

Pošlete nám program, který testuje, zda číslo na vstupu je prvočíslo a my vám pošleme robota!



Automatické vyhodnocení soutěže „test testu prvočíslnosti“

TEST PRVOČÍSELNOSTI

Instance: Zdrojový kód programu P .

Otázka: Přijímá P , právě když vstupní číslo je prvočíslo?

Problém **TEST PRVOČÍSELNOSTI** je nerozhodnutelný.

- **Sporem:** Máme program $\mathcal{T}$, který rozhoduje **TEST PRVOČÍSELNOSTI**
- Sestavíme program $\mathcal{V}$, který rozhoduje univerzální jazyk

$$L_u = \{ \langle M, x \rangle \mid x \in L(M) \}$$

Spor s nerozhodnutelností L_u

Rozhodujeme L_u voláním $\mathcal{T}$

Výpočet $\mathcal{V}$ se vstupem $\langle M, x \rangle$

- 1 Sestroj program P , který se řídí následujícím algoritmem

Výpočet P se vstupem n

- 1 Pust' $M(x)$
- 2 **if** M přijal **and** n je prvočíslo **then**
- 3 | **přijmi**
- 4 **else**
- 5 | **odmítni**

$L(P) = \begin{cases} \emptyset & \dots M(x) \text{ nepřijme} \\ \text{PRIME} = \{ \langle n \rangle \mid n \text{ je prvočíslo} \} & \end{cases}$

$\hookrightarrow M(x) \text{ přijme} \dots$

// Položíme dotaz programu $\mathcal{T}$

- 6 Pust' $\mathcal{T}(P)$ **if** $\mathcal{T}(P)$ přijal **then**
 - 7 | **přijmi**
 - 8 **else**
 - 9 | **odmítni**
-

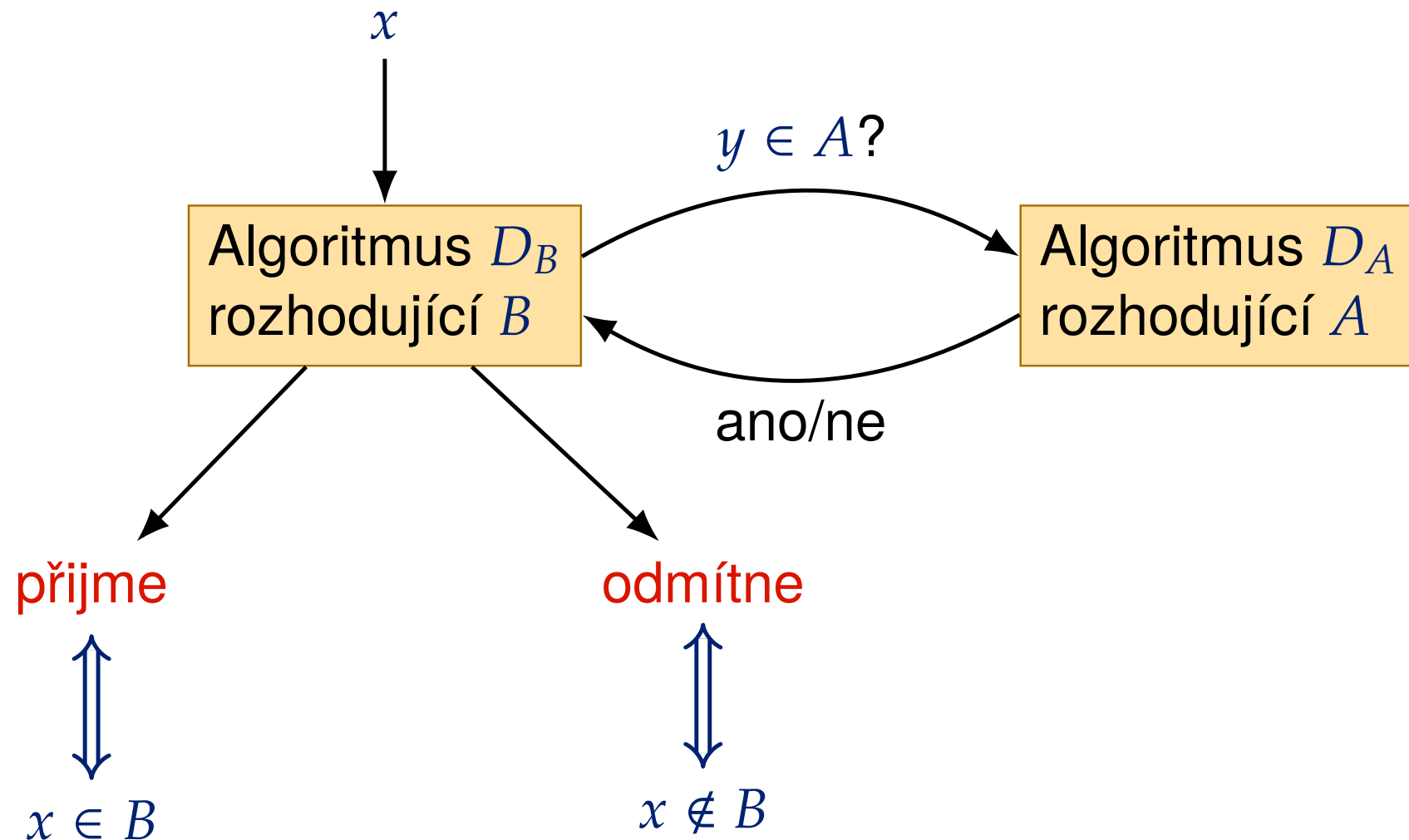
Jak ukazovat nerozhodnutelnost?

Chceme ukázat, že A je nerozhodnutelný jazyk.

- 1 Vybereme si jiný nerozhodnutelný jazyk B
 - například $B = L_u$
 - 2 Rozhodnutelnost $A \implies$ rozhodnutelnost B
 - Máme algoritmus D_A , který rozhoduje A
 - Popíšeme algoritmus D_B , který rozhoduje B
 - D_B může volat D_A jako podprogram
 - 3 Spor s nerozhodnutelností B
- $\implies A$ není rozhodnutelný

Turingovská převoditelnost (princip)

$B \leq_T A$ = „pomocí A umíme rozhodnout B “



Turingovská převoditelnost

Jazyk $B \subseteq \Sigma^*$ je **turingovsky převoditelný** na jazyk $A \subseteq \Sigma^*$ ($B \leq_T A$), pokud existuje algoritmus (Turingův stroj) D_B , který

- rozhoduje B
- může se ptát, zda „ $y \in A$ “ pro libovolný řetězec $y \in \Sigma^*$
 - Odpovídá D_A
 - Libovolný počet dotazů

- A rozhodnutelný $\implies B$ rozhodnutelný
- B nerozhodnutelný $\implies A$ nerozhodnutelný
- $A \leq_T \bar{A}$ pro každý jazyk A *$\hookrightarrow$ stačí otočit výstup*

Jazyk problému TEST PRVOČÍSELNOSTI

TEST PRVOČÍSELNOSTI

Instance: Zdrojový kód programu P .

Otázka: Přijímá P , právě když vstupní číslo je prvočíslo?

- Odpovídá mu jazyk

$$TP = \{\langle P \rangle \mid L(P) = \text{PRIME}\},$$

kde

$$\text{PRIME} = \{\langle p \rangle \mid p \text{ je prvočíslo}\}$$

Konstrukce programu $\mathcal{V}$ ukazuje, že $L_u \leq_T TP$.

Problém zastavení

PROBLÉM ZASTAVENÍ (HALTING PROBLEM)

Instance: Kód Turingova stroje M a vstup x .

Otázka: Zastaví se výpočet Turingova stroje M nad vstupem x ?

- Odpovídá mu jazyk

$$\text{HALT} = \{ \langle M, x \rangle \mid M(x) \downarrow \}$$

Věta

Jazyk HALT je částečně rozhodnutelný, ale není rozhodnutelný.

Částečná rozhodnutelnost HALT

$$\text{HALT} = \{ \langle M, x \rangle \mid M(x) \downarrow \}$$

- Plyne z existence univerzálního stroje $\mathcal{U}$
- HALT je přijímán následujícím strojem $\mathcal{H}$

Výpočet $\mathcal{H}$ se vstupem $\langle M, x \rangle$

1 Simuluj $M(x)$

2 **přijmi** $\rightarrow$ neznámí můj jestli reálně přijmu nebo zamítnu

Nerozhodnutelnost HALT

$$\text{HALT} = \{ \langle M, x \rangle \mid M(x) \downarrow \}$$

- **Sporem:** Necht' $\mathcal{H}$ je stroj rozhodující HALT
 - $\text{HALT} = L(\mathcal{H})$ a
 - $\mathcal{H}(\langle M, x \rangle) \downarrow$ pro každý Turingův stroj M a vstup x
- Popíšeme Turingův stroj M_u , který rozhoduje

$$L_u = \{ \langle M, x \rangle \mid x \in L(M) \}$$

Stroj M_u rozhodující L_u s pomocí HALT

Výpočet stroje M_u se vstupem $\langle M, x \rangle$

1 Sestroj Turingův stroj M' , který se řídí následujícím algoritmem

Výpočet M' se vstupem y

1 Pust' $M(y)$
2 **if** M zamítl **then**
3 vstup do nekonečné smyčky

Spojili jsme
nepřijetí a
zacyklení do
jednoho stroje

// Položíme dotaz jazyku HALT

2 **if** $\langle M', x \rangle \in \text{HALT}$ **then**

3 **přijmi**

4 **else**

5 **odmítni**

Nerozhodnutelnost $HALT$

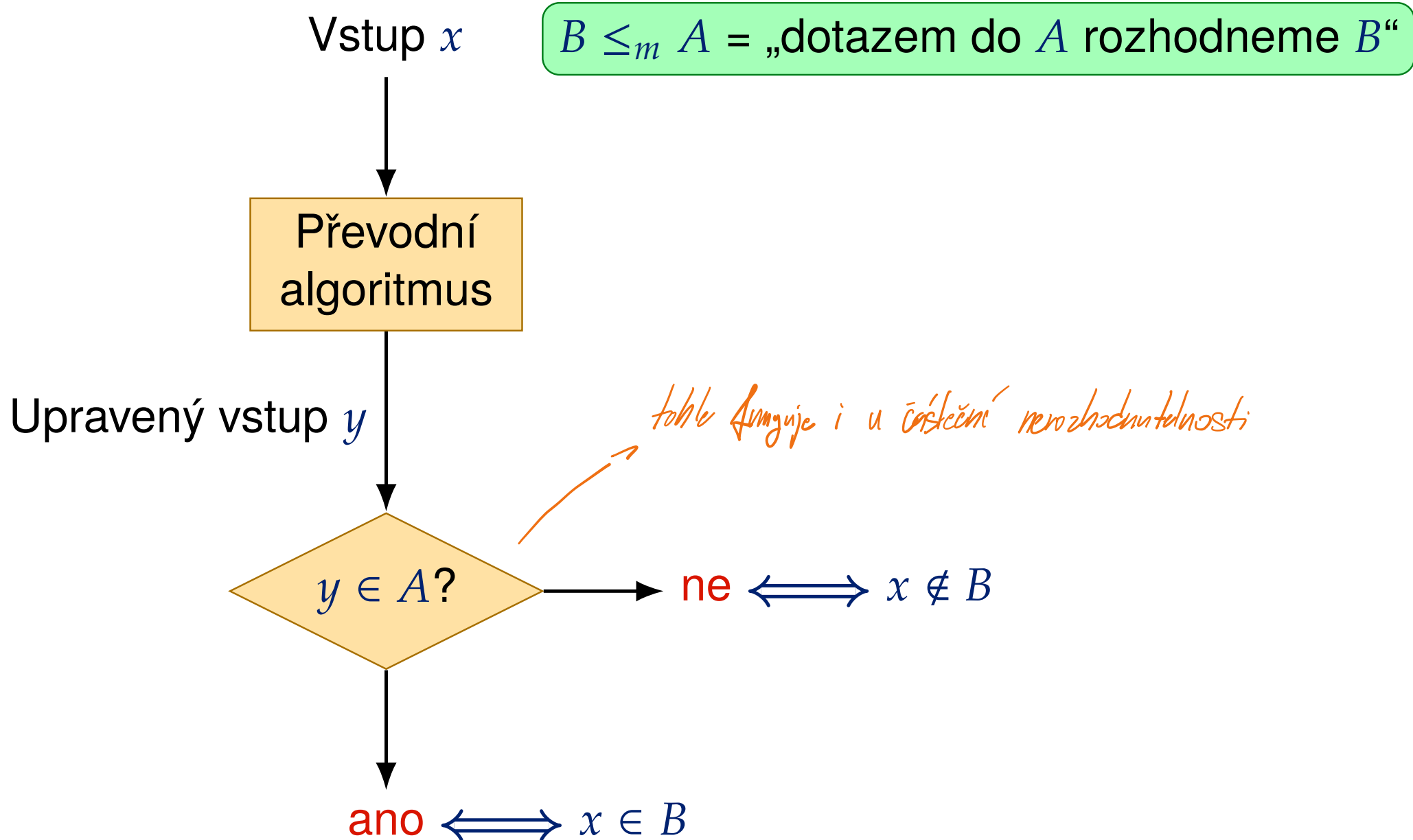
Ukázali jsme, že $L_u \leq_T HALT$

L_u není rozhodnutelný $\implies HALT$ není rozhodnutelný

- Program M_u je velmi specifický
 - Orákula $HALT$ se ptá jen jednou na konec
 - Odpověď dotazu $HALT$ je přímo odpovědí M_u

Program ukazuje, že L_u je m -převoditelný na $HALT$

m -převoditelnost (princip)

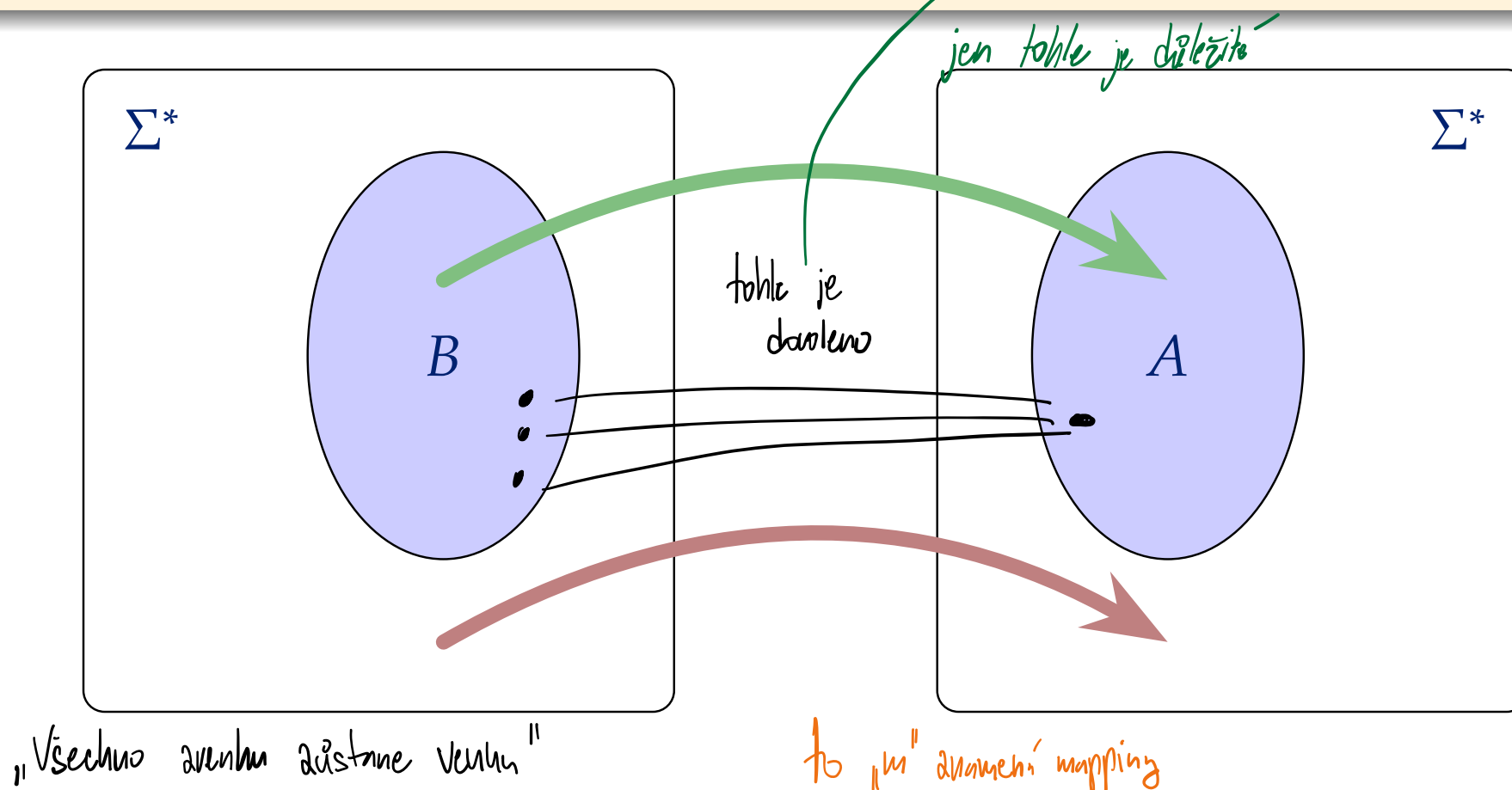


m -převoditelnost (definice)

Definice

Jazyk B je m -převoditelný na jazyk A , ~~překlad~~ existuje totální vyčíslitelná funkce f splňující $\Leftrightarrow$

$$(\forall x \in \Sigma^*) [x \in B \Leftrightarrow f(x) \in A]$$



m -převoditelnost (definice)

Definice

Jazyk B je m -převoditelný na jazyk A ($B \leq_m A$), pokud existuje totální vyčíslitelná funkce f splňující

$$(\forall x \in \Sigma^*)[x \in B \Leftrightarrow f(x) \in A]$$

- $\leq_m$ je reflexivní a tranzitivní relace (**kvaziuspořádání**).
- Pokud $A \leq_m B$ a B je (částečně) rozhodnutelný jazyk, pak totéž lze říct o A .

$$\underbrace{\overline{L_u}}_{\text{není částečně rozhodnutelný}} \not\leq_m \underbrace{L_u}_{\text{částečně rozhodnutelný}}$$

m -převoditelnost (reflexivita)

Lemma (Reflexivita m -převoditelnosti)

Pro každý jazyk A platí $A \leq_m A$

Důkaz.

- Identita $\text{id}(x) = x$ je totální algoritmicky vyčíslitelná funkce
- Pro každý řetězec $x \in \Sigma^*$ platí

$$x \in A \Leftrightarrow \text{id}(x) \in A$$



m -převoditelnost (tranzitivita)

Lemma (Tranzitivita m -převoditelnosti)

Pro každé jazyky A , B a C platí $A \leq_m B \wedge B \leq_m C \implies A \leq_m C$

Důkaz.

- $A \leq_m B$ pomocí funkce g
- $B \leq_m C$ pomocí funkce h
- Definujme funkci $f(x) = h(g(x))$
 - f je totální algoritmiicky vyčíslitelná funkce
- Pro každý řetězec $x \in \Sigma^*$ platí

$$x \in A \underbrace{\iff}_{A \leq_m B} g(x) \in B \underbrace{\iff}_{B \leq_m C} h(g(x)) \in C \underbrace{\iff}_{f(x)=h(g(x))} f(x) \in C$$

- $A \leq_m C$ pomocí funkce f

$\hookrightarrow$ $y \in B \iff h(y) \in C$



m -převoditelnost porovnává obtížnost

Lemma

Nechť B a A jsou dva jazyky, pro něž platí, že $B \leq_m A$.

- 1 A rozhodnutelný $\implies B$ je rozhodnutelný
- 2 A částečně rozhodnutelný $\implies B$ je částečně rozhodnutelný

- Předpokládejme, že M_A je TS, který přijímá/rozhoduje A
- Popíšeme TS M_B , který přijímá/rozhoduje B

Výpočet stroje M_B se vstupem x

```
1  $y \leftarrow f(x)$            //  $f$  ukazuje, že  $B \leq_m A$ 
2 Pust'  $M_A(y)$ 
3 if  $M_A$  přijal then
4   | přijmi
5 else
6   | odmítni
```

m -převoditelnost a nerozhodnutelnost

Lemma

Nechť B a A jsou dva jazyky, pro něž platí, že $B \leq_m A$.

- 1 A je rozhodnutelný $\implies B$ je rozhodnutelný
- 2 A je částečně rozhodnutelný $\implies B$ je částečně rozhodnutelný

Důsledek

Nechť B a A jsou dva jazyky, pro něž platí, že $B \leq_m A$.

- 1 B je nerozhodnutelný $\implies A$ je nerozhodnutelný
- 2 B není částečně rozhodnutelný $\implies A$ není částečně rozhodnutelný

Převod L_u na TP

- $L_u = \{\langle M, x \rangle \mid x \in L(M)\}$
- $TP = \{\langle P \rangle \mid L(P) = \text{PRIME}\}$

$$L_u \leq_m TP$$

Idea:

- Definujeme $f(\langle M, x \rangle) = \langle P \rangle$ tak, aby platilo, že

$$x \in L(M) \iff L(P) = \text{PRIME}$$

- Algoritmus počítající f provede konstrukci P
- Jde o konstrukci, kterou provádí program $\mathcal{V}$

$$L_u \leq_m \text{TP}$$

$$L_u = \{\langle M, x \rangle \mid x \in L(M)\}$$

$$\text{TP} = \{\langle P \rangle \mid L(P) = \text{PRIME}\}$$

K $\langle M, x \rangle$ definujeme program P takto

Výpočet P se vstupem n

Puť $M(x)$

if M přijal **and** n je prvočíslo **then**

 | **přijmi**

else

 | **odmítni**

- $M(x)$ přijme $\implies P$ provádí test prvočíselnosti
- $M(x)$ nepřijme $\implies L(P) = \emptyset$

Převod L_u na HALT

- $L_u = \{\langle M, x \rangle \mid x \in L(M)\}$
- $\text{HALT} = \{\langle M, x \rangle \mid M(x) \downarrow\}$

Lemma

$$L_u \leq_m \text{HALT}$$

Idea:

- Ke stroji M popíšeme stroj M' tak, aby pro každý vstup $x \in \Sigma^*$ platilo, že

$$M \text{ přijme } x \iff M'(x) \downarrow$$

- Definujeme $f(\langle M, x \rangle) = \langle M', x \rangle$
- Algoritmus počítající f provede transformaci M na M'

$$L_u \leq_m \text{HALT}$$

$$L_u = \{ \langle M, x \rangle \mid x \in L(M) \}$$

$$\text{HALT} = \{ \langle M, x \rangle \mid M(x) \downarrow \}$$

Ke stroji M definujeme stroj M' takto

Výpočet M' se vstupem y

Puť $M(y)$

if M zamítl **then** vstup do nekonečné smyčky

- Pro každý řetězec $y \in \Sigma^*$ platí:

$$y \in L(M) \iff M'(y) \downarrow$$

- Pro každou dvojici M a x platí

$$\langle M, x \rangle \in L_u \iff x \in L(M) \iff M'(x) \downarrow \iff \langle M', x \rangle \in \text{HALT}$$

$L_u \leq_m \text{HALT}$

- Funkce $f(\langle M, x \rangle) = \langle M', x \rangle$
 - je totální algoritmicky vyčíslitelná a
 - ukazuje, že $L_u \leq_m \text{HALT}$

Neprázdnost jazyka

PROBLÉM NEPRÁZDNÉHO JAZYKA

Instance: Kód Turingova stroje M

Otázka: Je $L(M) \neq \emptyset$?

$$NE = \{\langle M \rangle \mid L(M) \neq \emptyset\}$$

Věta

Jazyk NE je částečně rozhodnutelný ale není rozhodnutelný

- Částečnou rozhodnutelnost jsme již ukázali dříve
- Nerozhodnutelnost ukážeme tím, že $L_u \leq_m NE$

$$L_u \leq_m \text{NE}$$

$$L_u = \{\langle M, x \rangle \mid x \in L(M)\}$$

$$\text{NE} = \{\langle M \rangle \mid L(M) \neq \emptyset\}$$

K dvojici M a x definujeme Turingův stroj M'

Výpočet M' se vstupem y

Puť $M(x)$ // M' ignoruje svůj vstup!

if M přijal **then**

 | **přijmi**

else

 | **odmítni**

$$L(M') = \begin{cases} \Sigma^* & x \in L(M) \\ \emptyset & x \notin L(M) \end{cases}$$

$$L_u \leq_m \text{NE}$$

- Pro každý kód dvojice $\langle M, x \rangle$ platí

$$\langle M, x \rangle \in L_u \iff x \in L(M) \iff L(M') \neq \emptyset \iff \langle M' \rangle \in \text{NE}$$

- Funkce $f(\langle M, x \rangle) = \langle M' \rangle$
 - je totální algoritmicky vyčíslitelná a
 - ukazuje, že $L_u \leq_m \text{NE}$

Riceova věta

Riceova věta

Věta (Riceova věta)

Nechť C je třída částečně rozhodnutelných jazyků a položme

$$L_C = \{\langle M \rangle \mid L(M) \in C\}$$

Jazyk L_C rozhodnutelný, právě když je třída C *triviální*, tedy buď je prázdná nebo obsahuje všechny částečně rozhodnutelné jazyky.

- Nechť P je netriviální vlastnost jazyka
- Následující otázka je nerozhodnutelná

Splňuje $L(M)$ vlastnost P pro daný program M ?

Riceova věta (důsledky)

Nerozhodnutelnost následujících jazyků plyne z Riceovy věty

- $NE = \{\langle M \rangle \mid L(M) \neq \emptyset\}$
 - C je třída neprázdných částečně rozhodnutelných jazyků
- $Fin = \{\langle M \rangle \mid L(M) \text{ je konečný jazyk}\}$
 - C je třída konečných jazyků
- $Inf = \{\langle M \rangle \mid L(M) \text{ je nekonečný jazyk}\}$
 - C je třída nekonečných částečně rozhodnutelných jazyků
- $Dec = \{\langle M \rangle \mid L(M) \text{ je rozhodnutelný jazyk}\}$
 - C je třída rozhodnutelných jazyků
- $Reg = \{\langle M \rangle \mid L(M) \text{ je regulární jazyk}\}$
 - C je třída regulárních jazyků
- $TP = \{\langle M \rangle \mid L(M) = PRIME\}$
 - $PRIME = \{\langle p \rangle \mid p \text{ je prvočíslo}\}$
 - $C = \{PRIME\}$
- $Hello = \{\langle M \rangle \mid M \text{ přijímá řetězec Hello}\}$

Riceova věta (nepoužitelnost)

Věta (Riceova věta)

Nechť C je třída částečně rozhodnutelných jazyků a položme

$$L_C = \{ \langle M \rangle \mid L(M) \in C \}$$

*Jazyk L_C rozhodnutelný, právě když je třída C **triviální**, tedy buď je prázdná nebo obsahuje všechny částečně rozhodnutelné jazyky.*

NELZE použít na jazyky

$$L_u = \{ \langle M, x \rangle \mid x \in L(M) \}$$

$$\text{HALT} = \{ \langle M, x \rangle \mid M(x) \downarrow \}$$

$$\text{DIAG} = \{ \langle M \rangle \mid \langle M \rangle \notin L(M) \}$$

Riceova věta (důkaz)

Věta (Riceova věta)

Nechť C je třída částečně rozhodnutelných jazyků a položme

$$L_C = \{ \langle M \rangle \mid L(M) \in C \}$$

Jazyk L_C rozhodnutelný, právě když je třída C *triviální*, tedy buď je prázdná nebo obsahuje všechny částečně rozhodnutelné jazyky.

- Nechť třída C je prázdná
 - $\implies L_C = \emptyset$
 - $\implies L_C$ je rozhodnutelný
- Nechť třída C obsahuje všechny částečně rozhodnutelné jazyky
 - $\implies L_C = \Sigma^*$
 - $\implies L_C$ je rozhodnutelný

Riceova věta (důkaz)

- Dále předpokládáme, že C je netriviální
- Ukážeme, že
 - $L_u \leq_m \underline{L_C}$ pokud C neobsahuje prázdný jazyk
 - $L_u \leq_m \overline{L_C}$ pokud C obsahuje prázdný jazyk

Idea

Zobecnění převodu L_u na $TP = \{\langle P \rangle \mid L(P) = \text{PRIME}\}$.

Riceova věta (důkaz)

- Předpokládejme, že C neobsahuje prázdný jazyk
 - tedy $\emptyset \notin C$
- Zvolíme Turingův stroj M_1 takový, že $L(M_1) \in C$
 - Speciálně $L(M_1) \neq \emptyset$
- Popíšeme totální vyčíslitelnou funkci $f(\langle M, x \rangle) = \langle M' \rangle$ splňující

$$L(M') = \begin{cases} L(M_1) & x \in L(M) \\ \emptyset & x \notin L(M) \end{cases}$$

- Platí tedy

$$x \in L(M) \iff L(M') \in C \iff f(\langle M, x \rangle) = \langle M' \rangle \in L_C$$

- Funkce f ukazuje, že $L_u \leq_m L_C$

Riceova věta (důkaz)

K dvojici M a x definujeme Turingův stroj M'

Výpočet M' se vstupem y

Simuluj $M(x)$

if M přijal **then** // $x \in L(M)$

 Simuluj $M_1(y)$

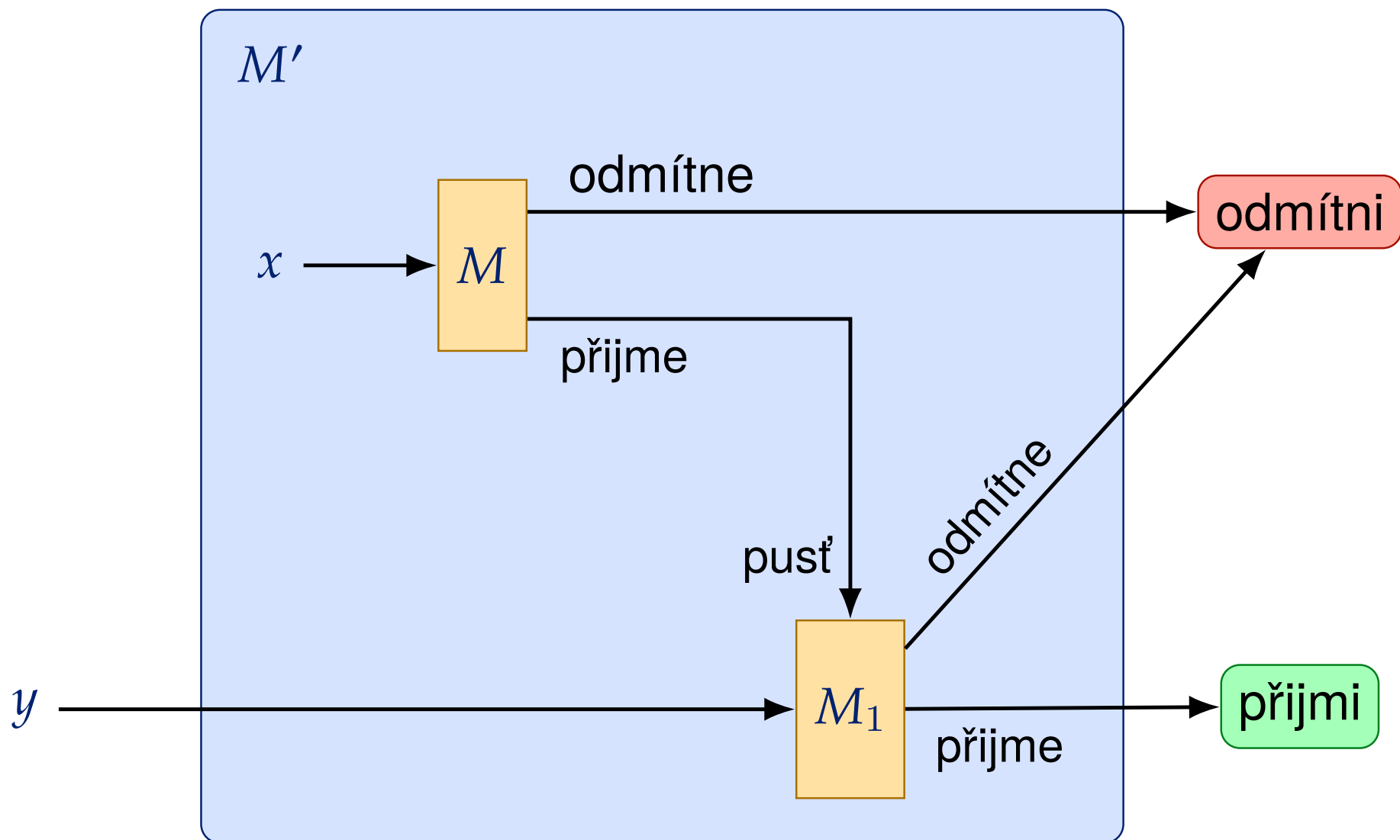
if M_1 přijal **then**

 přijmi

odmítni

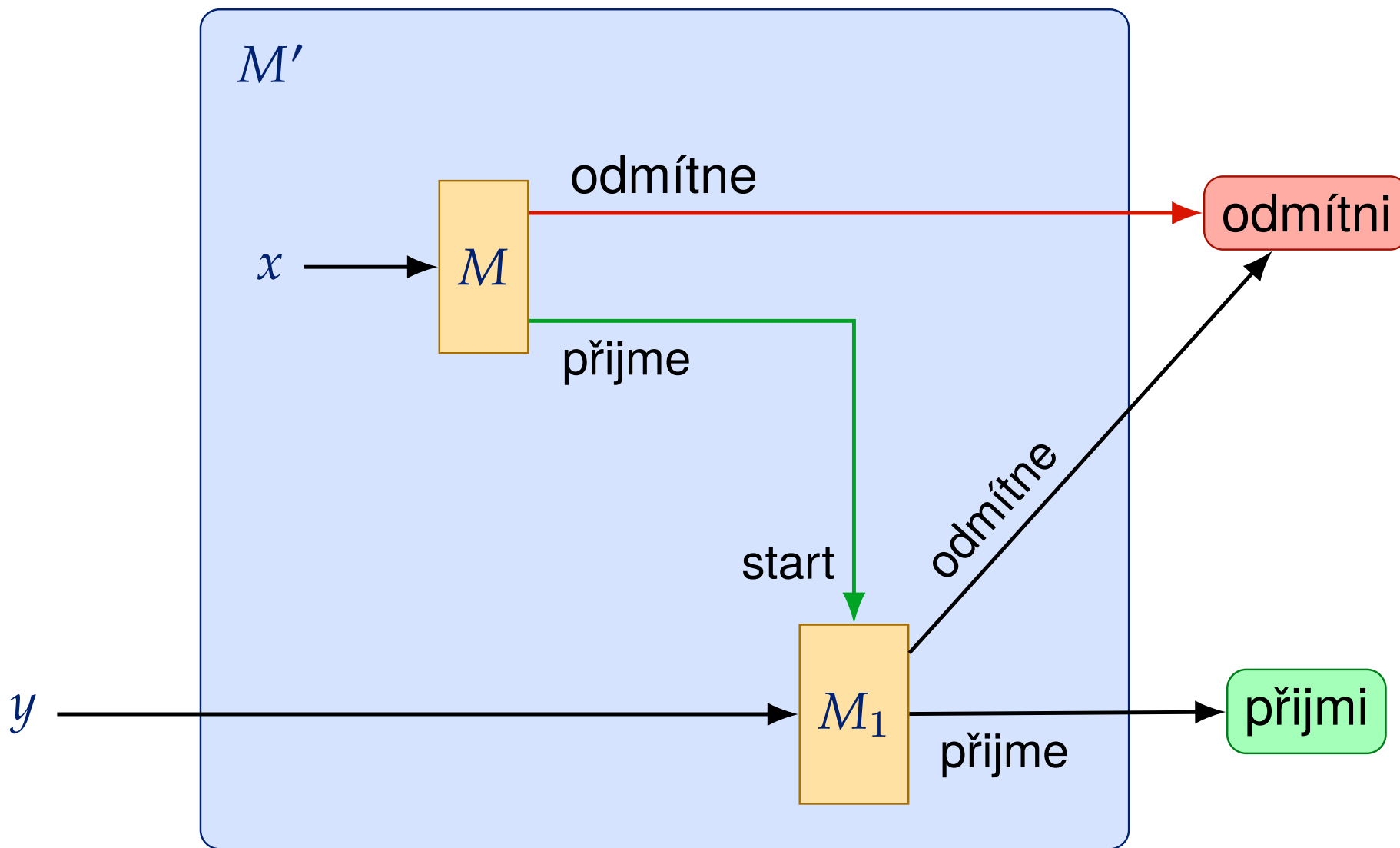
Riceova věta (důkaz)

K dvojici M a x definujeme Turingův stroj M'



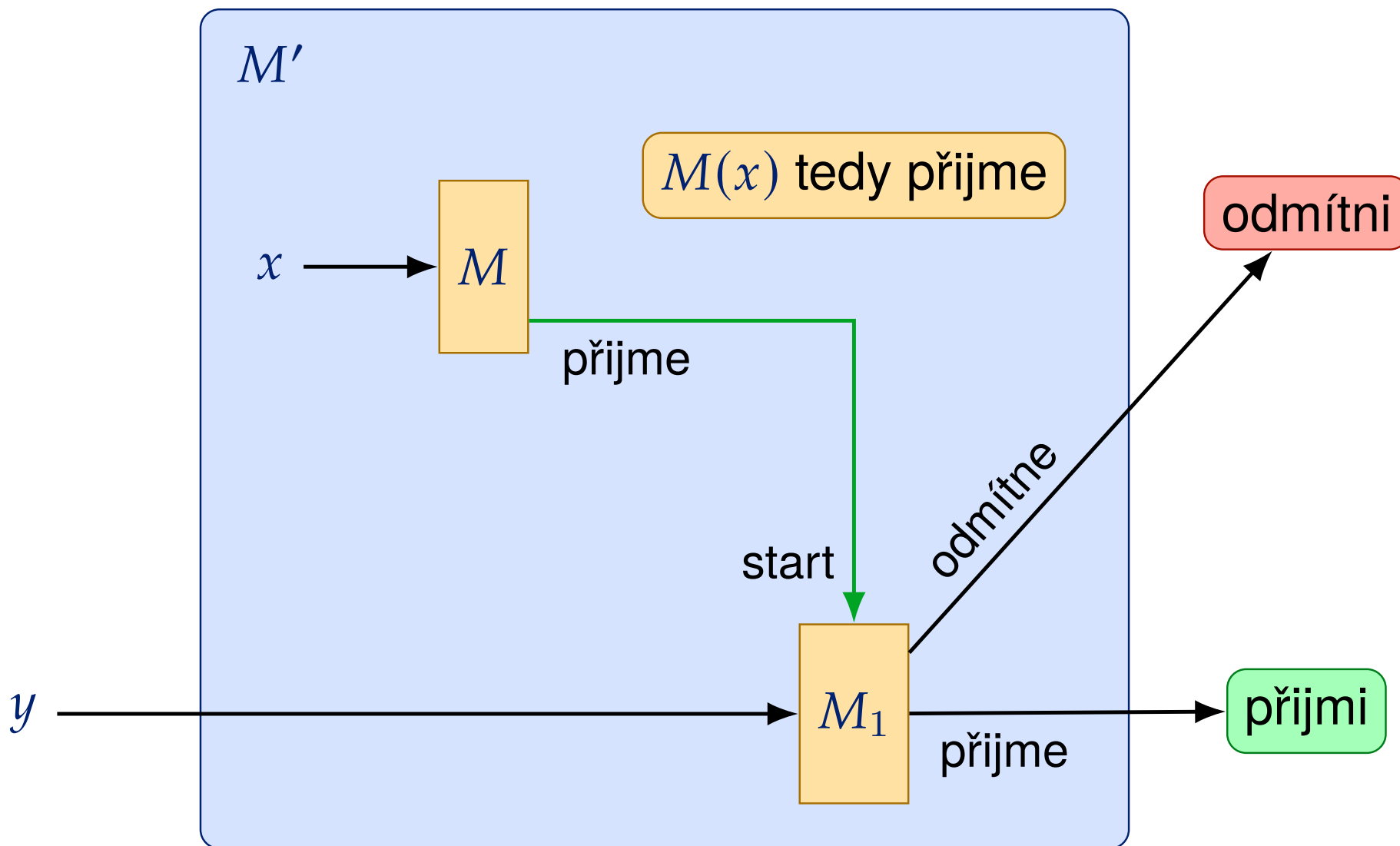
Riceova věta (důkaz, $x \in L(M)$)

Předpokládejme $x \in L(M)$



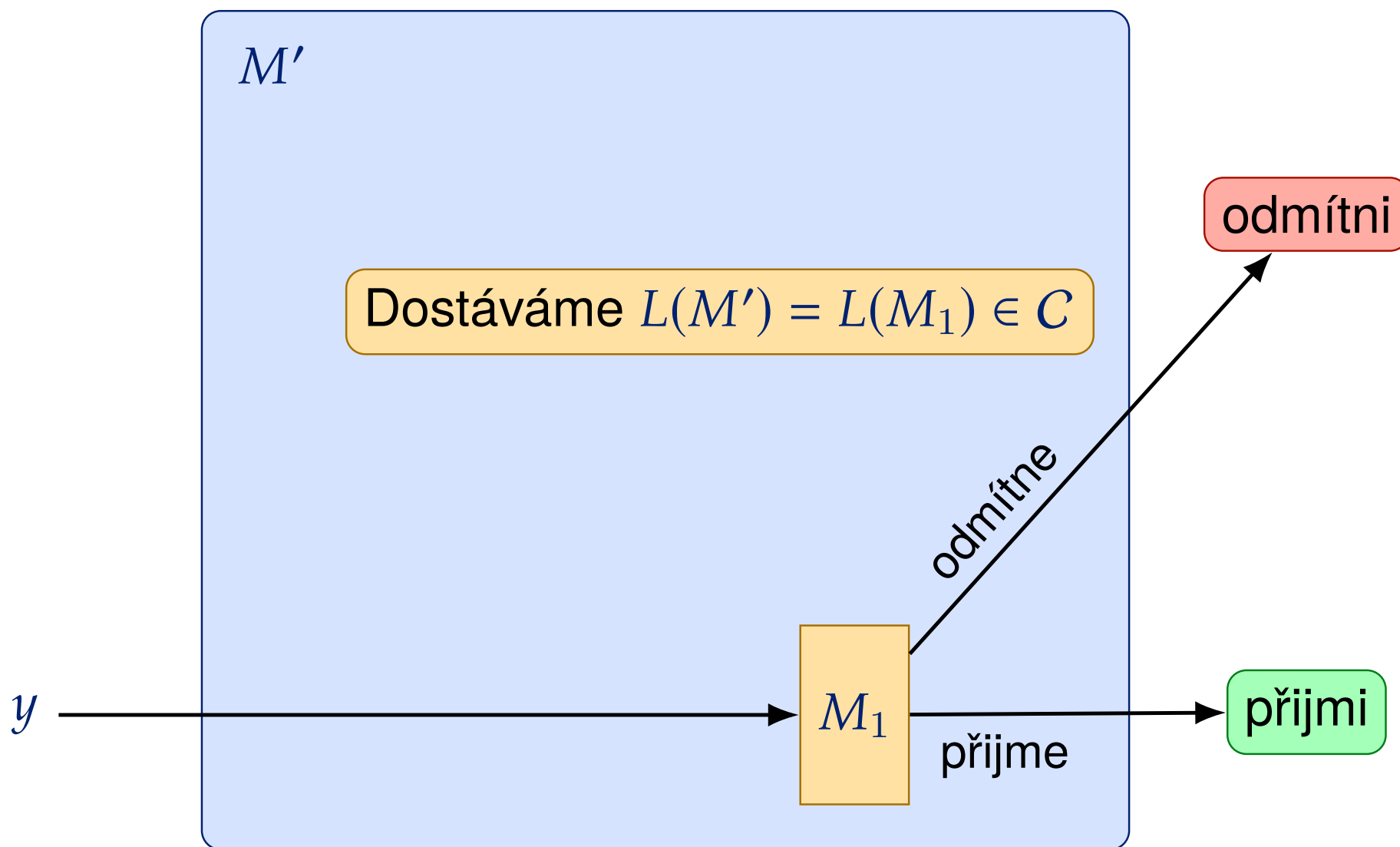
Riceova věta (důkaz, $x \in L(M)$)

Předpokládejme $x \in L(M)$



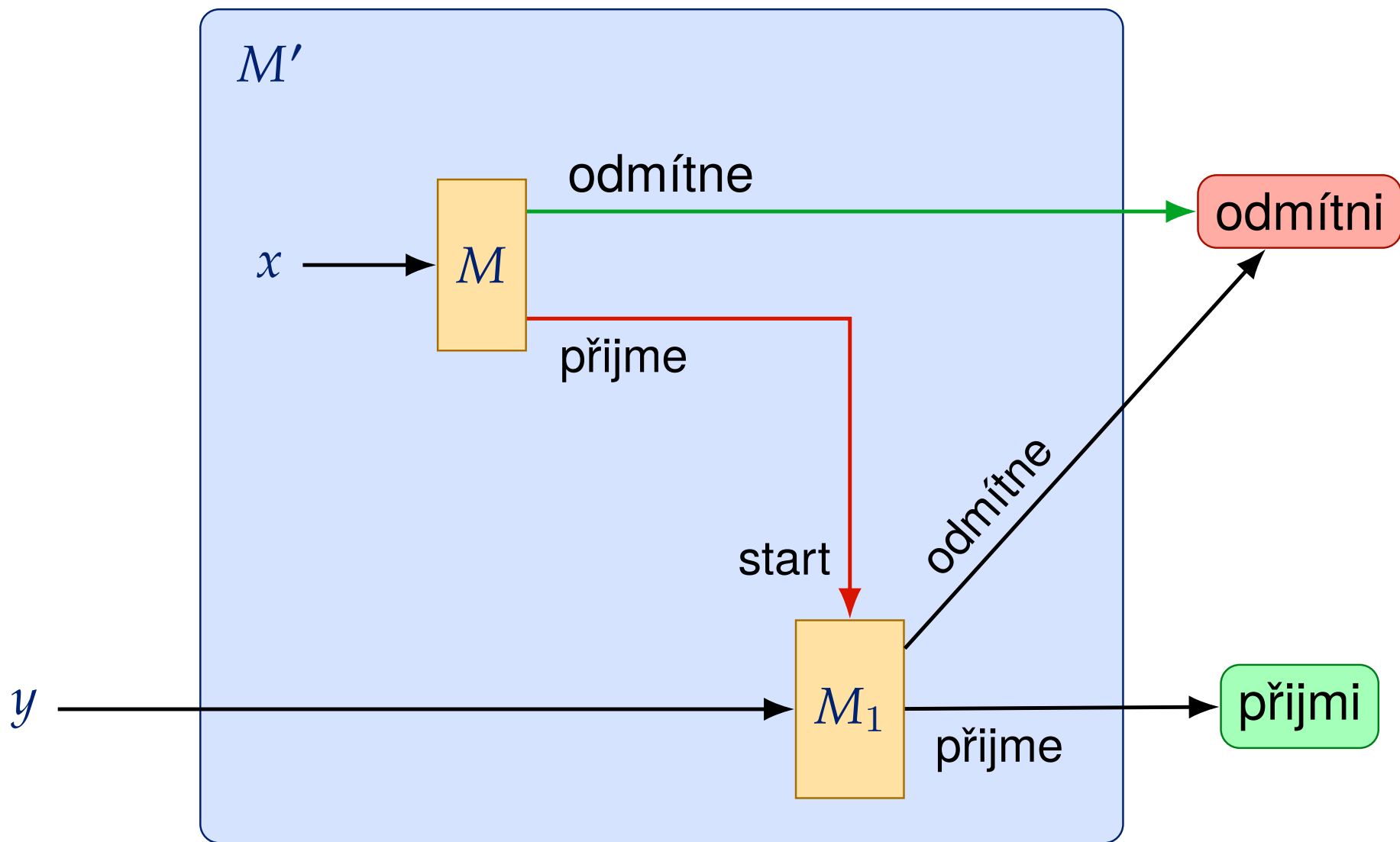
Riceova věta (důkaz, $x \in L(M)$)

Předpokládejme $x \in L(M)$



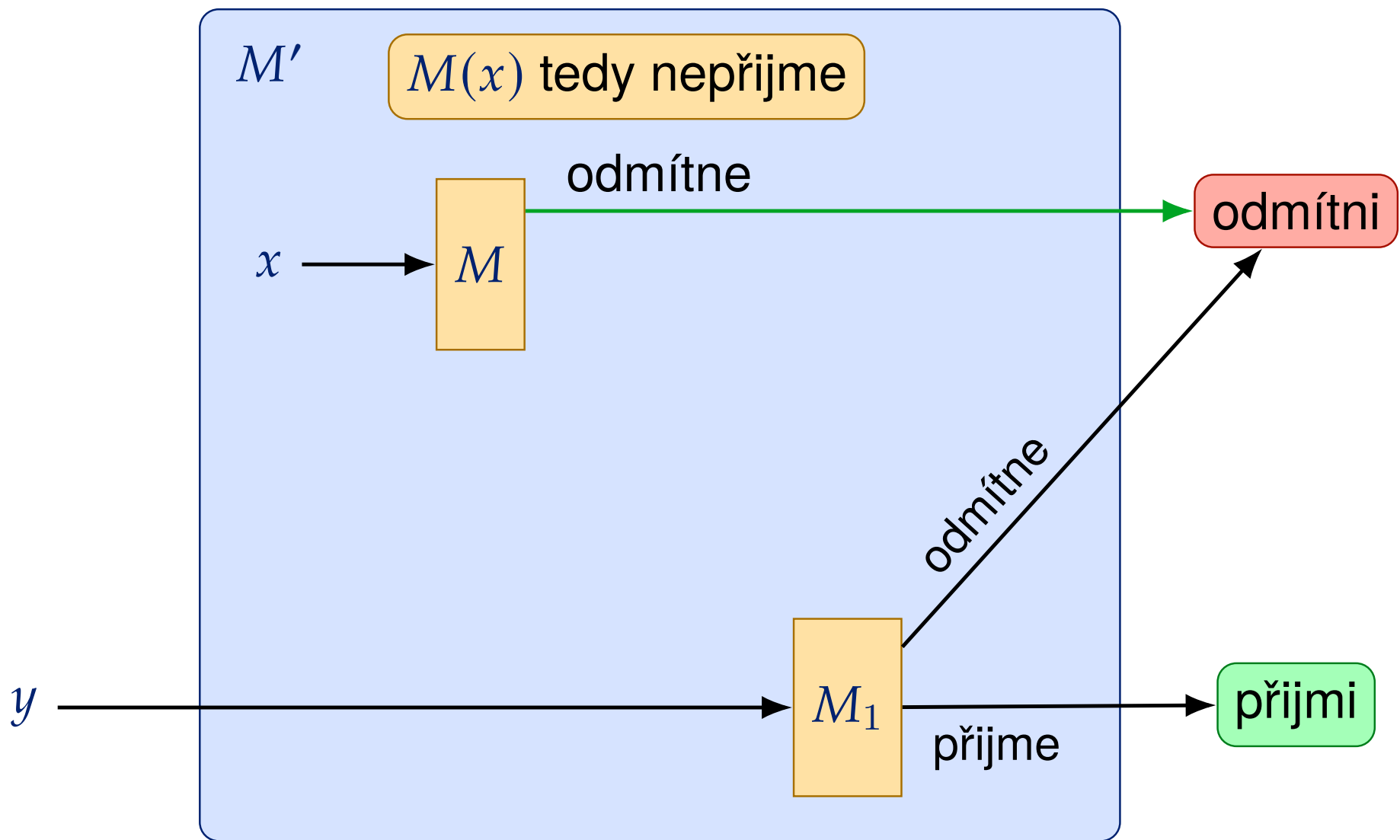
Riceova věta (důkaz, $x \notin L(M)$)

Předpokládejme $x \notin L(M)$



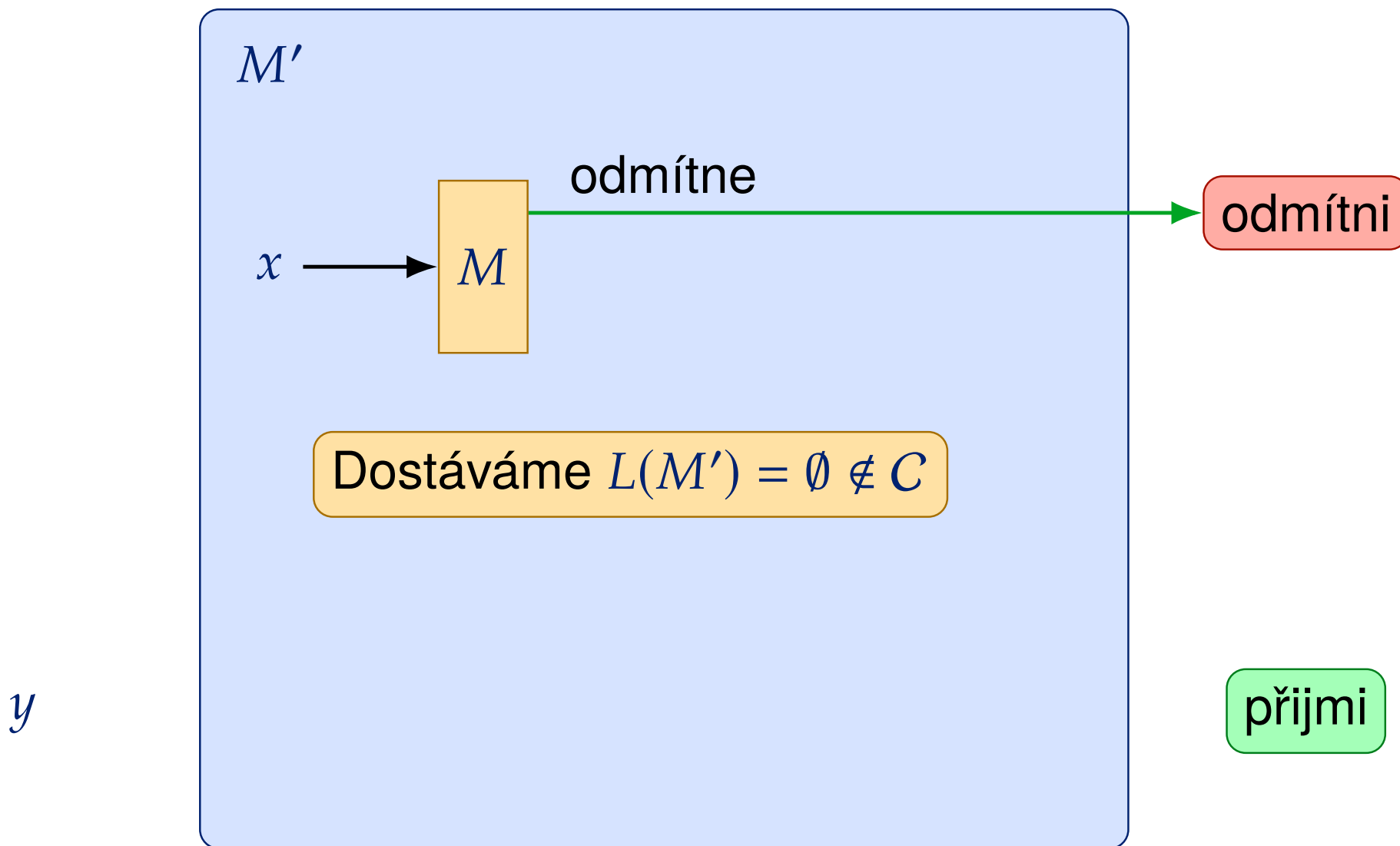
Riceova věta (důkaz, $x \notin L(M)$)

Předpokládejme $x \notin L(M)$



Riceova věta (důkaz, $x \notin L(M)$)

Předpokládejme $x \notin L(M)$



Riceova věta (důkaz, dokončení)

- $x \in L(M) \implies L(M') = L(M_1) \in C \implies \langle M' \rangle \in L_C$
- $x \notin L(M) \implies L(M') = \emptyset \notin C \implies \langle M' \rangle \notin L_C$
- Funkce $f(\langle M, x \rangle) = \langle M' \rangle$ je totální algoritmicky vyčíslitelná a platí

$$\langle M, x \rangle \in L_u \iff x \in L(M) \iff f(\langle M, x \rangle) = \langle M' \rangle \in L_C$$

- Dostáváme $L_u \leq_m L_C$
- Pokud $\emptyset \in C$
 - stačí vyměnit role C a doplňku C
 - tím ukážeme $L_u \leq_m \overline{L_C}$

Riceova věta (funkce)

Pomocí f_M označíme funkci, kterou počítá Turingův stroj M .

Věta (Riceova věta (funkce))

Nechť C je třída algoritmicky vyčíslitelných funkcí a položme

$$A_C = \{\langle M \rangle \mid f_M \in C\}$$

*Jazyk A_C rozhodnutelný, právě když je třída C **triviální**, tedy buď je prázdná nebo obsahuje všechny algoritmicky vyčíslitelné funkce.*

- Důkaz analogický verzi pro jazyky

Riceova věta (důsledky)

Následující jazyky jsou nerozhodnutelné dle Riceovy věty

- $\text{Tot} = \{\langle M \rangle \mid f_M \text{ je totální, tedy } \text{dom } f_M = \Sigma^*\}$
- $\text{Sum} = \{\langle M \rangle \mid M \text{ počítá součet dvou čísel}\}$
- $\text{Inc} = \{\langle M \rangle \mid f_M \text{ je rostoucí}\}$
- $\text{Zero} = \{\langle M \rangle \mid f_M(0) \downarrow = 0\}$
- $\text{ToUpper} = \{\langle M \rangle \mid M \text{ změní ve vstupu malá písmena na velká}\}$
- $\text{HW} = \{\langle M \rangle \mid M \text{ vypíše } \text{Hello, world} \text{ jako prvních } 12 \text{ znaků výstupu}\}$

Postův korespondenční problém

POSTŮV KORESPONDENČNÍ PROBLÉM

Instance: Množina „dominových kostek“ P :

$$P = \left\{ \left[\frac{t_1}{b_1} \right], \left[\frac{t_2}{b_2} \right], \dots, \left[\frac{t_k}{b_k} \right] \right\}$$

kde $t_1, \dots, t_k, b_1, \dots, b_k \in \Sigma^*$ jsou řetězce.

Otázka: Existuje **párovací posloupnost** $i_1, i_2, \dots, i_l$, kde $l \geq 1$
a $t_{i_1} t_{i_2} \dots t_{i_l} = b_{i_1} b_{i_2} \dots b_{i_l}$?

Věta (Bez důkazu)

POSTŮV KORESPONDENČNÍ PROBLÉM je nerozhodnutelný.