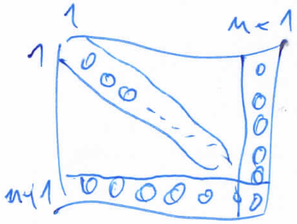
Ukažte, že třída P je uzavřena vůči iteraci jazyků

Idea

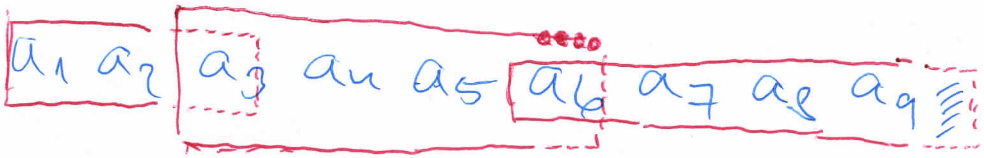
- Zvolme libovolný jazyk $L \in P$. / M ($L(M)=L$)
- K L musí existovat úplný DTS*, který přijímá (rozhoduje) L v polynomiálním čase $p(n) \in O(n^k)$, $k \geq 0$.
- Sestavíme úplný DTS M' takový, že $L(M') = L^*$.
- M' musí pracovat takto:
  - M' přijme, pokud je na vstupu ε.
  - M' si na pomocné pásce sestaví matici A rozměru $(n+1) \times (n+1)$ s Booleovskými vstupy s následující sémantikou:
    - $\forall\ 1 \leq i \leq n\ \forall\ 2 \leq j \leq n+1$:
      $A[i,j] = 1$ tehdy a jen tehdy, když $a_i a_{i+1} \dots a_{j-1}$, kde $a_1 \dots a_n$ je vstupní řetězec M', patří do L ($a_i \dots a_{j-1} \in L$).
      Jinak $A[i,j] = 0$.
    - $\forall\ 1 \leq j \leq n+1: A[n+1,j] = 0$
      $\forall\ 1 \leq i \leq n+1: A[i,1] = 0$



$\forall\, 1 \leq i \leq n+1: A[i,i] = 0$

- Pro vyplnění matice musí M' použít simulaci M na pomocných páskách. Bude ho spouštět $O(n^2)\times$. Složitost celé této operace je tedy $O(n^2) \cdot T(n)$, neboli polynomiální.
- M' vypočte, např. pomocí Warshallova alg., tranzitivní uzávěr R^+ relace R reprezentované maticí A.
- M' ~~odpoví~~ přijme, pokud $(1, n+1) \in R^+$, což znamená, že $a_1 \dots a_n$ lze pokrýt na sebe navazujícími podřetězci z L. Jinak odmítne. $\square$

Př.



$n = 9$, vezmu podřetězce dané indexy

$(1,3), (3,6), (6,10)$

$a_1a_2 \in L$, $a_3a_4a_5 \in L$, $a_6a_7a_8a_9 \in L$ $\Rightarrow$ $a_1 \dots a_9 \in L^*$

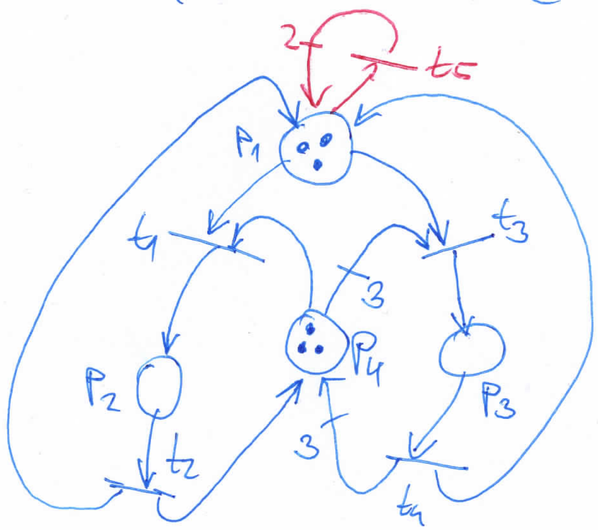
- Ukázat uzavřenost třídy NP vůči $\cup, \cap, \cdot, {}^*$.

Idea: Lze využít z důkazů uzavřenosti P vůči daným operacím, které je možno v případě $\cdot$ a $*$ zjednodušit tak, že stroj uhádne dělení vstupního řetězce na podřetězce – není zapotřebí iterace.

Petriho sítě

- Příklad P/T Petriho sítě modelující systém čtenářů a písařů pro max. 3 čtenáře.

doplnit



Síť:
- není bezpečná (konkrétně $M_0(p_1) > 1$).
- omezená: $\forall M \in [M_0\rangle \; \forall p: M(p) \leq 3$.
- není striktně konzervativní:
  - t_3 odebere 4 značky a vyprodukuje 1.

- Je značení $\begin{array}{cccc} p_1 & p_2 & p_3 & p_4 \\ (1, & 2, & 0, & 1) \end{array}$ dosažitelné?

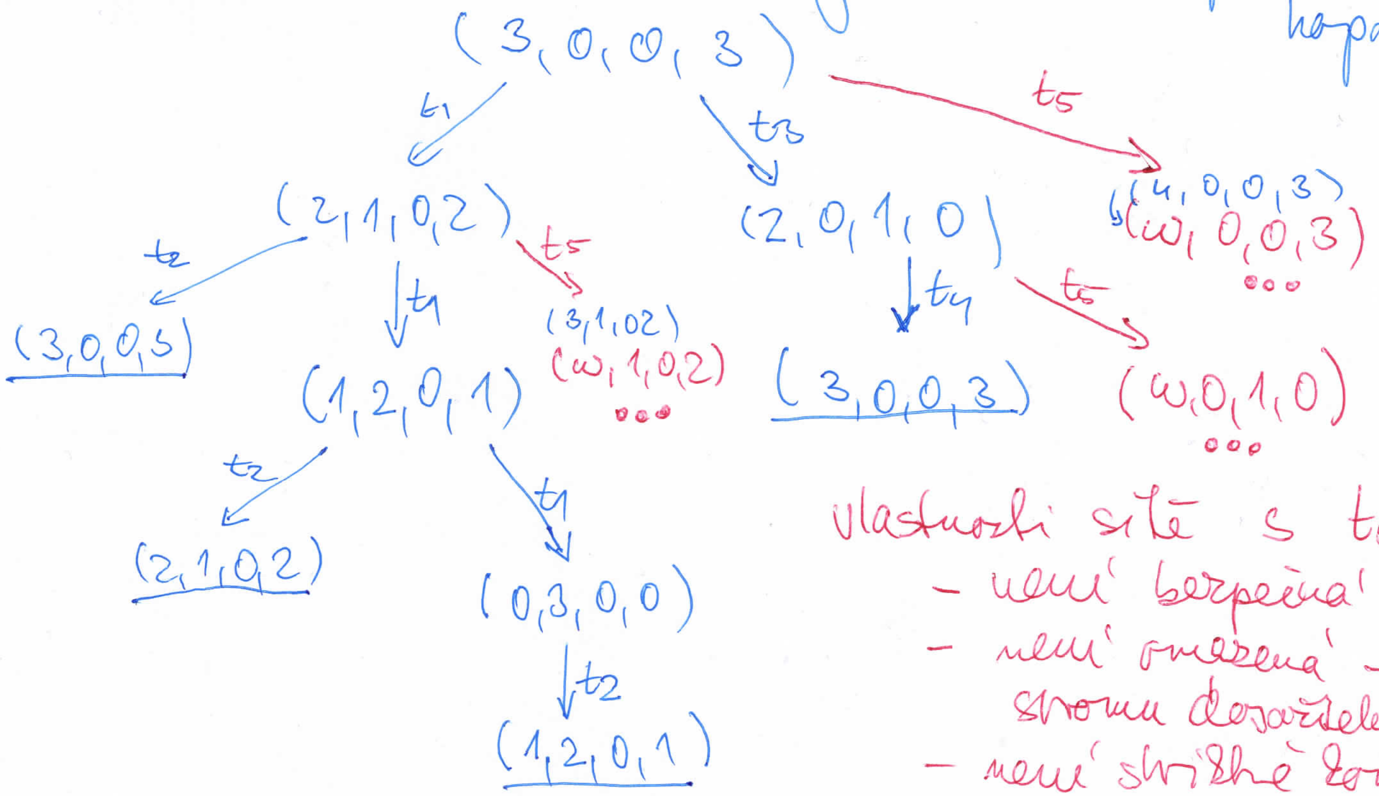
  ANO: $(3,0,0,3)\,[t_1\rangle\,(2,1,0,2)$
  $[t_1\rangle\,(1,2,0,1)$

- Je značení $(1,2,0,0)$ dosažitelné?
  NE

- Je značení $(1,2,0,0)$ pokrytelné?
  ANO, protože $(1,2,0,1)$ je dosažitelné.

• t_1 odebere 2 značky, vytvoří 1
- je konzervativní – lze vzít váhu 2 pro značky v p_2 a váhu 4 pro značky v p_3.
- síť je živá.

- <u>Konstrukce stromu dosažitelných značení</u> (pro síť bez omezení kapacit)



obecně může dojít ke ztrátě informace.

vlastnosti sítě s t_5:
- není bezpečná
- není omezená – viz ω ve stromu dosažitelnosti
- není striktně konzervativní!

– Je značení (1,2,0,1) dosažitelné:

ANO – obecně pro značení, které nejsou ve stromě přímo zachyceny, jsou pouze pokryty, nelze pomocí str. dos. značení rozhodnout.

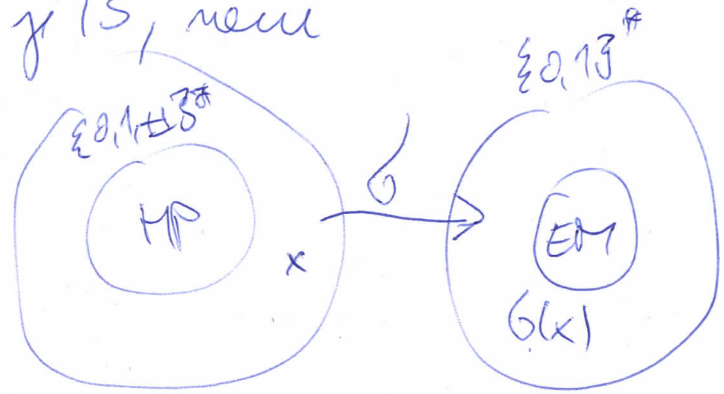
– není konzervativní (s ohledem na w)

– je živá

– může ale obecně rozhodnout pomocí stromu dos. značení!

– Na rozdíl od dosažitelnosti, pokrytelnost pomocí uvedené techniky lze rozhodovat.

- Ukážeme, že problém určení, zda $\varepsilon \in L(M)$, kde M je TS, není rozhodnutelný.



IDEA (náznak):

- Problém lze charakterizovat jazykem
  $EM = \{\langle M\rangle \mid \varepsilon \in L(M)\}$
- Lze užít redukci z $HP = \{\langle M\rangle \# \langle w\rangle \mid M$ je TS, který zastaví na $w\}$.
- redukce σ bude přiřazovat řetězci $x \in \Sigma_{HP} = \{0,1,\#\}$ řetězec $\sigma(x) \in \{0,1\}^*$ tak, že $\sigma(x) = \langle M_x\rangle$, kde M_x je kód TS fungující následovně:
  - M_x smaže svůj vstup.
  - M_x ověří, zda x je korektně zformovaná instance HP. Pokud ne odmítne.
  - Pokud ano, pak $x = x_1 \# x_2$ a M_x bude simulovat TS s kódem x_1 na řetězci s kódem x_2.
  - Pokud simulace skončí, přijme. Jinak cyklí.

- Redukci δ lze snadno implementovat úplným TS (viz důkaz nerozhodnutelnosti neprázdnosti).
- Redukce zachovává členství v jazyce, konečně:

	HP	EM
a) x nesprávně zform. instance MP	NE	$L(M_x) = \emptyset \to$ NE $\varepsilon \notin \emptyset$
b) $x = x_1 \# x_2$ a TS s kódem x_1 zastaví na vstupu s kódem x_2	ANO	$L(M_x) = \{0,1\}^* \to$ ANO
c) $x = x_1 \# x_2$, ale TS s kódem x_1 nezastaví na řetězci s kódem x_2	NE	$L(M_x) = \emptyset \Rightarrow$ NE

- Tedy $x \in MP \iff \delta(x) \in EM$. $\triangleleft$

- Jazyk $L \subseteq \Sigma^*$ nazveme netriviální, pokud $L \neq \emptyset$ a $L = \Sigma^*$.
- Ukažte, že problém rozhodování, zda jazyk $L(M)$ TS M není triviální, není ani částečně rozhodnutelný.
  - Uvedený problém lze charakterizovat jazykem $NT = \{\langle M \rangle \mid M \text{ je TS takový, že } L(M) \neq \emptyset \wedge L(M) \neq \Sigma^*\}$
  - POZOR, nelze užít stejnou redukci jako výše z co-MP.

	co-MP	NT
a) x není dobře zf.	NE ($x \notin$ co-MP)	$L(M_x) = \emptyset$ $\langle M_x \rangle \notin NT$ (NE)
b) $x = x_1 \# x_2$ a TS s kódem x_1 zastaví na vstupu s kódem x_2	$x \notin$ co-MP NE	$L(M_x) = \Sigma^{\#}$ $\langle M_x \rangle \notin NT$ (NE)
c) $x = x_1 \# x_2$ a TS s kódem x_1 nezastaví na vst. s kódem x_2	$x \in$ co-MP ANO	$L(M_x) = \emptyset$ $\langle M_x \rangle \notin NT$ (NE)

NELZE!

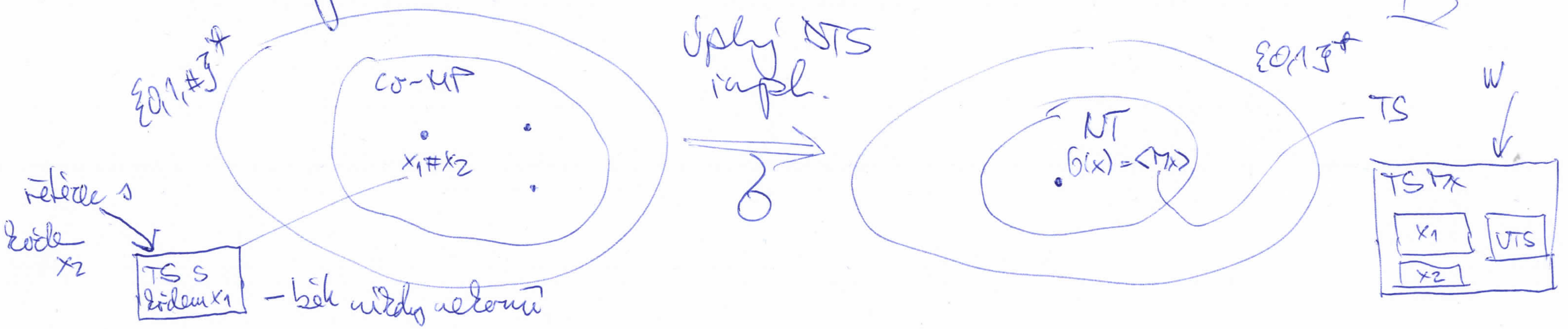
<u>Lze postupovat následovně:</u>

- Redukce σ bude redukovat $x \in \{0,1,\#\}^{\#}$ na $\sigma(x) = \langle M_x \rangle$, kde M_x je TS pracující následovně:
  1. M_x se podívá na vst. pásku, je-li ε, přijme.
  2. Jinak smaže vstup a zapíše na něj x.
  3. Jestliže x není dobře zformovaná instance co-MP, přijme.
  4. Jinak spustí simulaci TS s kódem x_1 na vstupu s kódem x_2 (kde $x = x_1 \# x_2$). Pokud skončí, přijme. Jinak cyklí.

- Redukci δ lze implementovat úplným TS, což lze ukázat podobně jako např. u redukce z MP na NEP.
- S ohledem na členství v jazycích co-MP a NT, lze učinit následující pozorování:

	co-MP	NT
a) pokud x není správně zform. instance co-MP	$x \notin$ co-MP ("NE")	$L(M_x)=\Sigma^*$ $\langle M_x\rangle \notin NT$ ("NE")
b) $x = x_1\#x_2$ a TS s kódem x_1 na vstupu s kódem x_2 skončí	$x \notin$ co-MP ("NE")	$L(M_x)=\Sigma^*$ $\langle M_x\rangle \notin NT$ ("NE")
c) $x = x_1\#x_2$ a TS s kódem x_1 na vstupu s kódem x_2 nezastaví	$x \in$ co-MP ("ANO")	$L(M_x)=\{\varepsilon\}$ $\langle M_x\rangle \in NT$ ("ANO")

Tedy $x \in$ co-MP $\Leftrightarrow \langle M_x\rangle = \delta(x) \in NT$. $\triangleright$



– Dokažte NP-úplnost 3SAT (víte-li, že SAT je NP-úplný).

Idea (námět)

a) členství v NP : triviální, protože každá instance 3SAT je instancí SAT. Tedy z toho, že SAT je v NP přímo plyne, že 3SAT je v NP.

b) NP-těžkost

- lze ukázat pomocí polynomiální redukce ze SAT.
- Je zapotřebí být schopen převést každou klausuli $L_1 \vee L_2 \vee \dots \vee L_n$ pro $n > 3$ na konjunkci klausulí o 3 literálech.
- k tomu lze užít převod na následující formuli

$$(L_1 \vee L_2 \vee X_3) \wedge$$
$$\wedge (X_3 \Rightarrow L_3 \vee X_4) \wedge$$
$$\wedge (X_4 \Rightarrow L_4 \vee X_5) \wedge$$

Zde $X_3, X_4, \dots, X_{n-1}$ jsou nové booleovské prom.

$\dots$

$\wedge\ (x_{n-1} \Rightarrow L_{n-1} \vee L_n)$

- Výše uvedené je přitom ekvivalentní formuli

$$(L_1 \vee L_2 \vee x_3) \wedge$$
$$\wedge\ (\neg x_3 \vee L_3 \vee x_4) \wedge$$
$$\wedge\ (\neg x_4 \vee L_4 \vee x_5) \wedge$$
$$\dots$$
$$\wedge\ (\neg x_{n-1} \vee L_{n-1} \vee L_n).$$

- Tato formule je ve formátu 3SAT.
- Snadno ji vyprodukuje úplný DTS pracující v polynom. čase – stačí jeden průchod formulí (ve vhodném řádu), přičemž si na pomocné pásce generuje pomocí čítače nové proměnné.
- Lze se navíc snadno přesvědčit, že vzniklá formule sice není logicky ekvivalentní, ale je tzv. „equisatisfiable" (splnitelná

tehdy a jen tehdy, když je splnitelná pův. formule !. Z toho plyne zachování členství v jazyce.

□