JAiO II (lecture 5)

The star height problem and distance automata.

L - regular language

$sh(L)$ = the smallest nesting depth of $*$ in a regular expression e defining L

$$sh\ (a^* b^*)^* = 1$$

Input: a regular language L, $n \in \mathbb{N}$

Question: $sh(L) \leq n$?

Question: how to compute $sh(L)$?

Question: how to compute the optimal expression ?

A brief history:

- posed in 1963 (Eggan)
- solved in 1988 (Hashiguchi) distance automata (198
- ....
- Kirsten (2005) PSPACE-complete
- Bojańczyk (2015)

starheight problem

↓ (tutorials)

boundedness

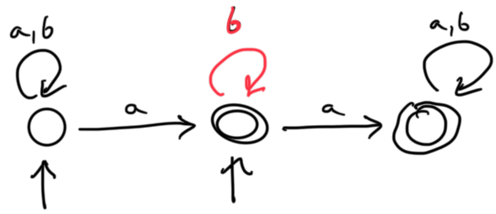
limitedness of distance automata

↓

solving ω-regular games

Distance automaton = NFA with a distinguished subset of costly transitions

Example:



Cost of a run = nr of costly transitions used

cost of an input word w = $\begin{cases} \text{the least cost of an } \underline{\text{accepting}} \text{ run over } w \\ \text{or } \infty \text{ if there is no accepting run} \end{cases}$

$\text{cost} : \Sigma^* \longrightarrow \mathbb{N} \cup \{\infty\}$

<u>The limitedness (boundedness) problem</u>:

Input: a distance automaton

Question: $\exists\, n \in \mathbb{N}.$ every input word has cost $< n$?

(limitedness $\Rightarrow$ universality)

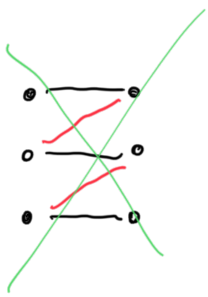
distance automaton A

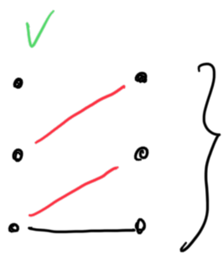
$\downarrow$

ω-regular game $\underline{G_\omega}$ $\quad (G_m,\ m \in \mathbb{N} \cup \{\omega\})$

<u>Players</u>:

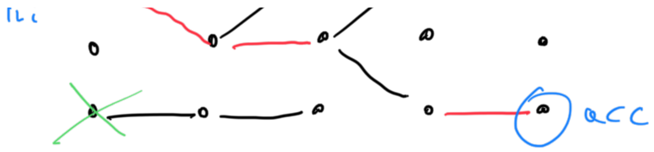
- Input player chooses a letter $\underline{a \in \Sigma}$
- Automaton player chooses a subset of <u>transitions of A</u> (over a)





in-degree ≤ 1

A play:



Winning condition for Automaton:

[Acc] after every round, some accepting state is reached from some initial one

[cost<n] every (possibly infinite) path costs $< n$ $\left[n = \omega \Rightarrow \text{cost is finite} \right.$

Lemma: The following conditions are equivalent:

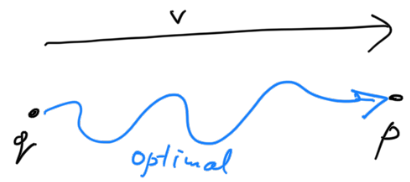
[Lim] A satisfies limitedness

[$\exists n$] $\exists n \in \mathbb{N}$. player Automaton wins G_n

[ω] player Automaton wins G_ω

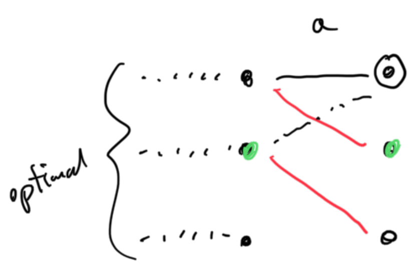
1) [Lim] $\Rightarrow$ [$\exists n$]

$\exists n$



Automaton's strategy: use optimal runs with cost $< n$ 1

starting in initial state 2



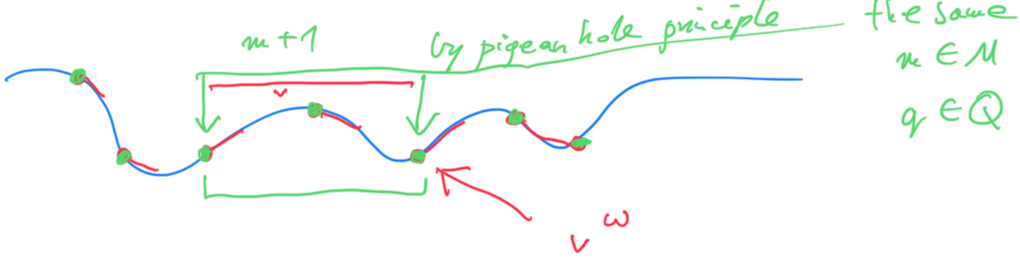
2) [ω] $\Rightarrow$ [$\exists n$] ?

$\rightarrow$ winning finite-memory strategy for Automaton in G_ω

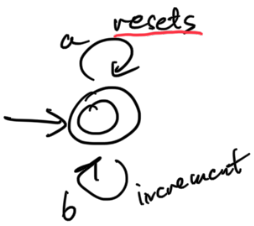
$\rightarrow \delta : M \times \Sigma \rightarrow M \times \Gamma$ ← subsets of transitions of A

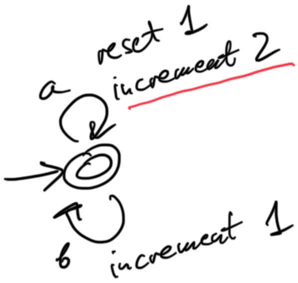
Claim: σ is winning in G_m, $m = |M| \times |\underbrace{Q}_{\text{states of } A}|$

Suppose that Automaton produces, using σ_1 a path of cost $\geq m$



Extensions of distance automata:





cost : $\Sigma^* \to \mathbb{N} \cup \{\infty\}$?

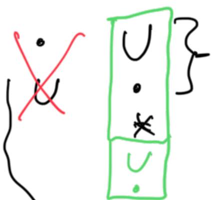
nested $1 < 2$

operations on this

reset this

$sh(L) = 0 \equiv L$ finite

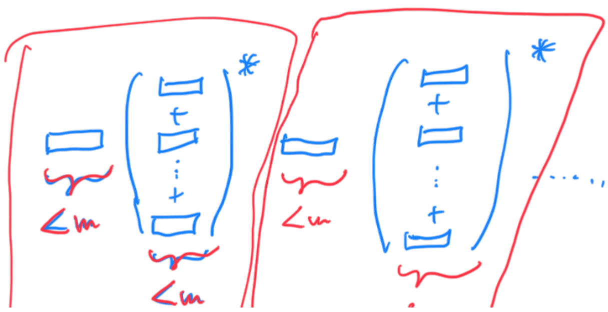
$sh(L) \leq 1$: $\exists\, m \in \mathbb{N}$



$(L \cup K) \cdot (M \cup N) =$

$= L \cdot M \cup L \cdot N \cup \ldots$

L as a finite union of languages



$< m$

$< m$

$w_1 \cdot t_1 \cdot w_2 \cdot t_2 \cdot \ldots \cdot w_n \cdot$

<u>Generalized</u> star height problem $\cup, \cdot, ^*,$ complement

$gsh(L) = 0 \equiv L$ is star-free $\equiv L$ FO-definable

<u>Open problem</u>: is there a regular language L s.t.

$$gsh(L) > 1 \; ?$$