HISTORY DETERMINISM

Lecture 1: Introduction to History-determinism through examples

Reference: When a little non-determinism goes a long way: An introduction to history-determinism

Udi Boker & Karoliina Lehtinen

ACM Siglog News, Volume 10, Issue 1 (2023)

Plan for the lecture:

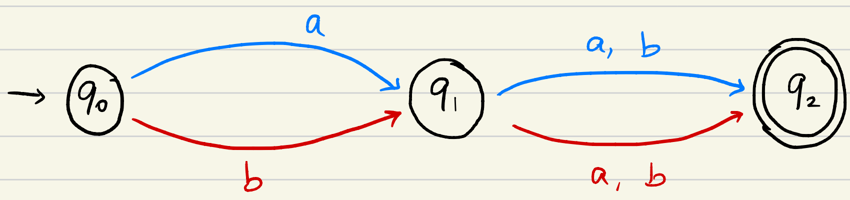
- Example 1
- Example 2
- Example 3
- Example 4
- Why is HD interesting? What questions can one ask about HD automata?

Suggested reading: Introduction of the reference given above

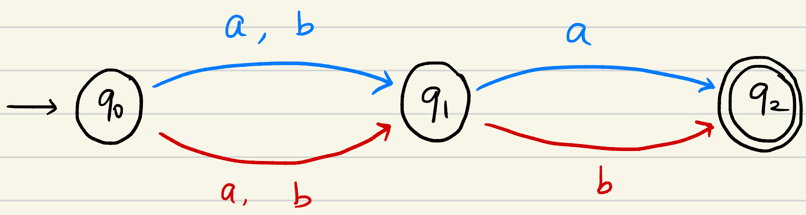
Example 1: From Karoliina's talk titled the same as the ACM Siglog paper

Consider these two automata:

$\mathcal{A}_0$:



$\mathcal{A}_0'$:



Acceptance condition: Both transitions should be of the same colour

Language of both automata: $\{aa, ab, ba, bb\}$

- Both the automata are non-deterministic. In $\mathcal{A}_0$, the non-determinism appears at q_1, whereas in $\mathcal{A}_0'$, it appears in q_0.

- In $\mathcal{A}_0$ the non-determinism at q_1 can be resolved as follows:

- If word read so far is 'a', take the blue transition. Else, take the red

This strategy to resolve non-determinism gives accepting runs on all the words in the language.

- In $\mathcal{A}_0'$, the non-determinism depends on the "future"
  - if the second letter is guessed to be 'a', take the blue transition
  - else take the red transition

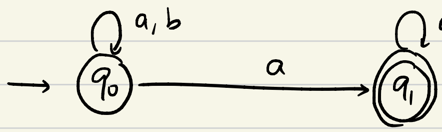
Automaton $\mathcal{A}_0$ is said to be history-deterministic.

* Intuitively, non-determinism in an HD automaton can be resolved by looking at the past alone.

Example 2:

Consider the following automaton over infinite words.

$\mathcal{A}_1$:



coBüchi acceptance: eventually run should remain in q_1.

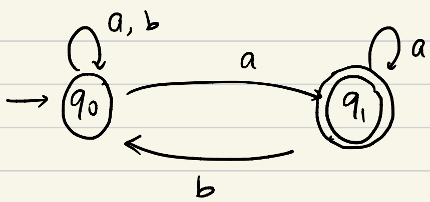
- 1. What is the language of $\mathcal{A}_1$?
   - words with finitely many b's
   - that is, $(a+b)^* a^\omega$

- 2. Is $\mathcal{A}_1$ deterministic ?
   - No. At q_0, on 'a' there are two possible choices.

- 3. How many accepting runs are possible on the word a^ω ?
   - Infinitely many. Can go from q_0 to q_1 after the i^{th} 'a', for any i

Consider a different automaton accepting the same language:

$\mathcal{A}_1'$:



coBüchi acceptance: eventually run should remain at q_1

$\mathcal{A}_1'$ is also non-deterministic. However, there is a strategy to get an accepting run for words in the language.

$\boxed{\text{At } q_0\text{, on 'a', always go to } q_1}$

- 1. Will this strategy give an accepting run for words in the language?

Yes. In any word in the language, there will be a trailing a^ω suffix. Here the run remains at q_1.

- 2. Will the strategy make a word not in the language get accepted?

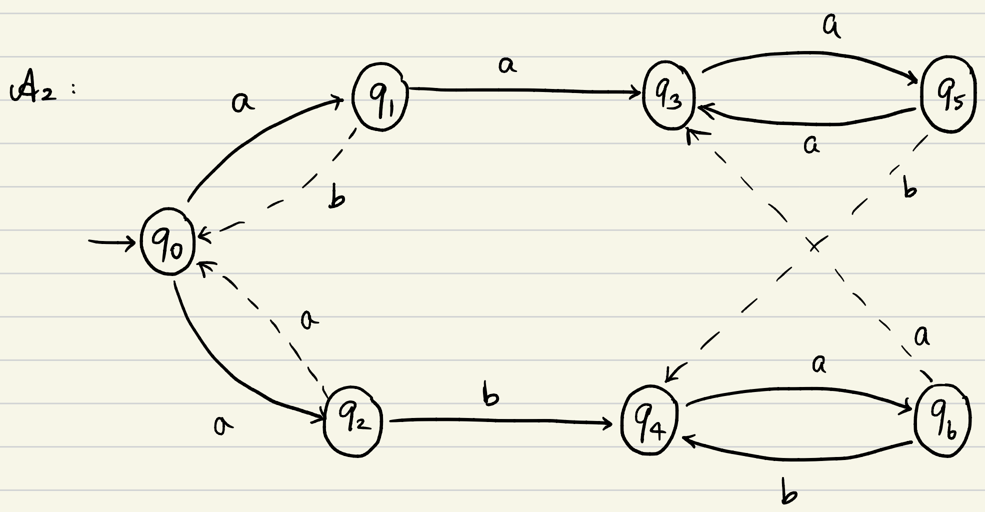
No! The strategy only carves a specific run among all possible runs. For words that are rejected, all runs are rejecting.

We say that automaton $\mathcal{A}_1'$ is HD.

Note: Automaton $\mathcal{A}_1$ is not HD. Later in the course, we will see how to identify HD automata.

We will see that $\mathcal{A}_1$ is not HD. We cannot find 'such a resolver' for $\mathcal{A}_1$.

Example 3:



- Dotted transitions are rejecting
- coBüchi acceptance condition : eventually, no rejecting transitions should appear in the run

Language of $\mathcal{A}_2$: $(ab + aa)^{*}[a^{\omega} + (ab)^{\omega}]$

- $\mathcal{A}_2$ is non-deterministic. Non-determinism appears only at state q_0.

Here is a resolver for $\mathcal{A}_2$:

– at q_0 alternate between q_1 and q_2

Exercise: show that this resolver yields an accepting run for every word in the language.

This resolver is a witness that $\mathcal{A}_2$ is HD.

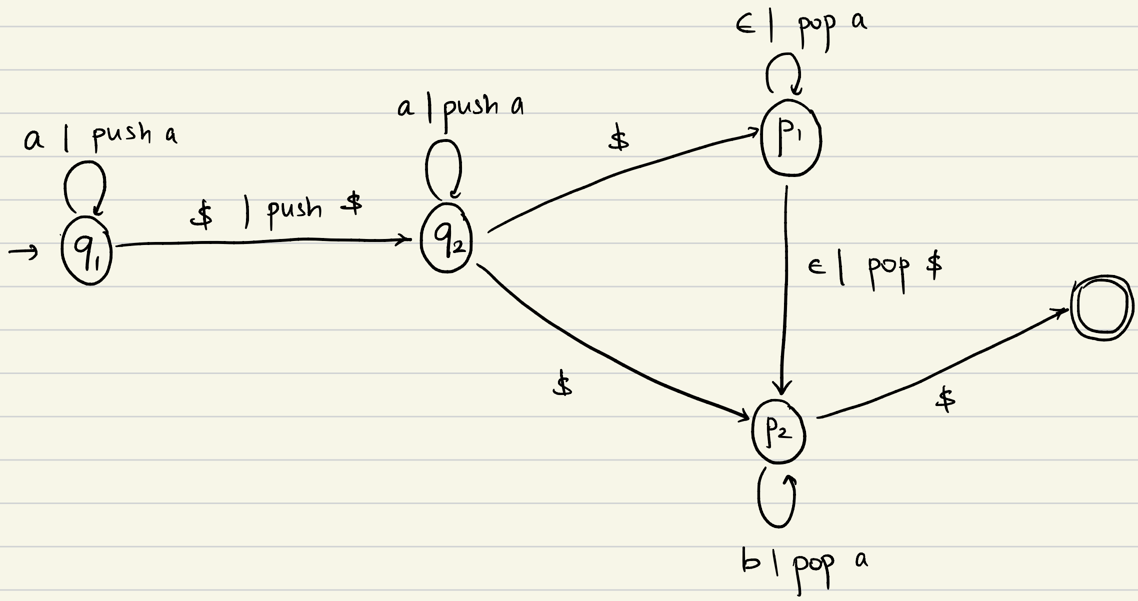
Example 4:

We now consider a context-free language.

$$L_3 = \{ \; a^i \$ a^j \$ b^k \$ \mid k \le i \text{ or } k \le j \}$$

This language cannot be accepted by a deterministic pushdown automaton.

Here is a non-deterministic PDA for L_3:



- The automaton first pushes $a^i \$ a^j$ onto the stack
- At q_2 it non-deterministically chooses to compare the b's with either i (by going to p_1) or j (by going to p_2)

Here is a resolver for this automaton:

- at q_2, go to p_1 if $i \geq j$, else goto p_2

This preserves the language. The above resolver is a witness to the automaton being history-deterministic.

Why is HD interesting?

- Non-deterministic automata are more succinct, whereas deterministic automata are better for algorithmic reasoning, with more efficient algorithms.

  History-determinism aims to extract some of the best from both worlds.

- In some contexts, HD is more expressive than deterministic automata.

- Sometimes, HD does not add expressive power, but is exponentially more succinct.

- HD automata are also well suited for solving games with ω-regular winning conditions (next lecture)

- For some models, HD automata give 'decidable inclusion' whereas language inclusion is undecidable with non-deterministic automata

What questions can one ask about history determinism?

In this course we will focus on ω-automata (coBüchi, Büchi, Parity)

- When are HD automata more expressive?
- When are they succinct?
- How to identify HD automata?