HISTORY DETERMINISM

Lecture 9: Characterizing History-Determinism in coBüchi automata

---

What we have: A co-Büchi automaton $\mathcal{A}$ in which Eve wins 1-token game from everywhere.

To show: $\mathcal{A}$ is HD

↳ i.e., Eve wins the letter / HD game on $\mathcal{A}$.

Question 1: We know Eve wins $G1^{\mathcal{A}}(q_0 ; q_0)$. Can this winning strategy be used in the HD game somehow?

In the HD game:

- → Adam chooses letters
- → Eve chooses a run.

At round i, Eve is at q_i

- → Adam chooses a_i
- → Eve chooses a transition $q_i \xrightarrow{a_i} q_{i+1}$

In $G1^{\mathcal{A}}(q_0; q_0)$

At round i, Eve is at q_i, Adam is at p_i

- → Adam chooses a letter a_i
- → Eve chooses transition $q_i \xrightarrow{a_i} q_{i+1}$
- → Adam chooses transition $p_i \xrightarrow{a_i} p_{i+1}$

In $G1$, Adam is also required to produce a run, whereas in the HD game Adam does not need to do so.

So Adam has more flexibility in the HD game.

→ A strategy σ of $G1$ cannot directly be employed for the HD game.

Question 2: We have a stronger hypothesis on $\mathcal{A}$
↓
Eve wins 1-token game on $\mathcal{A}$ from everywhere.

This implies $\mathcal{A}$ is semantically deterministic. Can we use this property?

Semantic determinism (SD):

A parity automaton $\mathcal{A}$ is said to be semantically deterministic (SD) if for all transitions:

$$q \xrightarrow{a} q_1 \qquad q \xrightarrow{a} q_2$$

$$L(\mathcal{A}, q_1) = L(\mathcal{A}, q_2)$$

↓
set of words with an acc. run starting from q_1.

- SD-ness implies that every non-deterministic choice leads to language equivalent states.

Lemma: Let $\mathcal{A}$ be a parity automaton. If Eve wins 1-token game from everywhere on $\mathcal{A}$, then $\mathcal{A}$ is SD.

Proof: Consider $q \xrightarrow{a} q_1$, $q \xrightarrow{a} q_2$

q_1, q_2 are co-reachable states.

$\therefore$ Eve wins $G1^{\mathcal{A}}(q_1 ; q_2)$ and $G1^{\mathcal{A}}(q_2 ; q_1)$

$\therefore$ $L(\mathcal{A}, q_2) \subseteq L(\mathcal{A}, q_1)$ and $L(\mathcal{A}, q_1) \subseteq L(\mathcal{A}, q_2)$

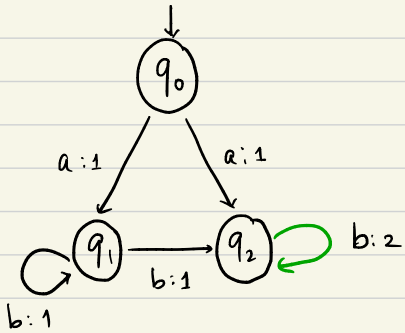
$\Rightarrow$ $L(\mathcal{A}, q_1) = L(\mathcal{A}, q_2)$

Let us get back to the coBüchi automaton $\mathcal{A}$ that we have.

Since Eve wins from everywhere, $\mathcal{A}$ is semantically deterministic.

Question: Does this mean Eve can choose any transition in the HD game?

No! Consider the following example.



$$L(q_1) = L(q_2) = b^\omega$$

All non-deterministic choices lead to q_1 & q_2. $\therefore$ the above automaton is SD.

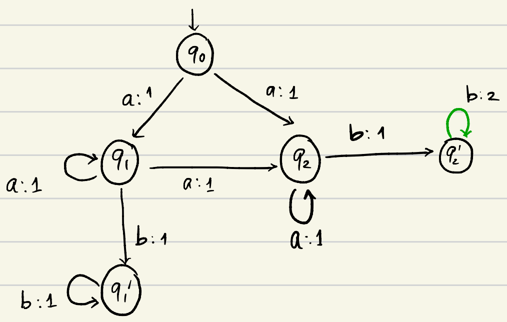
However Eve cannot arbitrarily choose transitions in the HD game.
→ for instance on ab^ω, $q_0 \longrightarrow q_1 \longrightarrow q_1 \rightarrow \cdots$ is rejecting run.

Therefore: Even when $\mathcal{A}$ turns out to be SD, Eve has to carefully choose her strategy.

The DAG of all runs:

We will now observe some properties about an object which represents the set of all runs of $\mathcal{A}$ on a word w.

We start with an example.

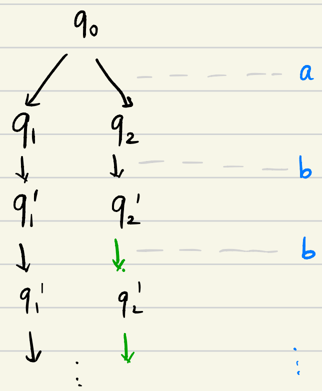


Automaton

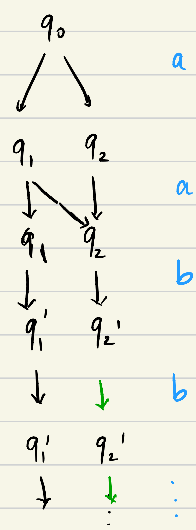
→ Not an SD automaton (why?)

Used only for illustrating DAG.

For ab^ω, the DAG would be:



For aab^ω, the DAG would be:



Intuitively the DAG represents a subset construction on the word.

→ The "width" of the DAG is bounded by the number of states of $\mathcal{A}$.

Formal definition of the DAG:

Given an automaton $\mathcal{A}$ and a word $w \in \Sigma^\omega$,

the DAG $\mathrm{Runs}_w^{\mathcal{A}}$ is defined as follows.

- Vertices: for every prefix u of w (including ε), we have nodes:

  (u, q) for all q s.t.

  $q_0 \xrightarrow{u} q$ in $\mathcal{A}$ (there is a run from q_0 to q on u)

- Edges: for all $a \in \Sigma$,

  $(u, q) \xrightarrow{a:c} (ua, q')$

  if $q \xrightarrow{a:c} q'$ is a transition of $\mathcal{A}$.

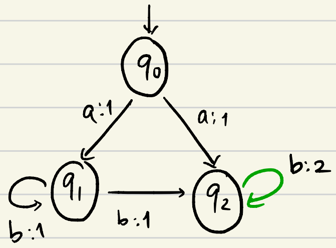
  Here 'c' denotes the colour of the transition.

Properties of the DAG:

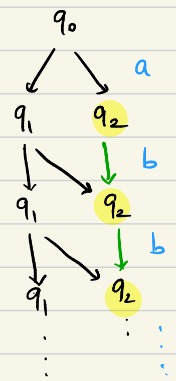
Consider a word $w \in \mathcal{L}(\mathcal{A})$

→ A node (u, q) of $Runs_w^{\mathcal{A}}$ is good if there exists an infinite sequence of edges with colour 2 starting from (u, q)

For example: consider



+ word ab^ω



Highlighted nodes are good.

Observation: When $w \in \mathcal{L}(\mathcal{A})$, from every node (u,q) in $Runs_w^{\mathcal{A}}$ there exists a path to a good node.

Suppose $w = uw'$, denote w' as $u^{-1}w$.

Since $\mathcal{A}$ is SD, $u^{-1}w$ is in the language of all states reached on u.

Hence there is an acc. run from q on $u^{-1}w$.

Therefore there should be a path to a good node.
(because of ∞ Büchi acceptance)

Back to Eve's strategy in the letter game:

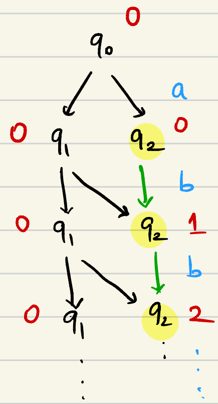
Eve's strategy should therefore be to hit a good node. But notice that Eve does not have the entire word upfront. By looking at the word seen so far, she has to ensure that eventually a good node is hit and a green run is taken from there!

Eve's strategy in the letter game (a simple attempt)

Our first step is to attach a number to every node (u,q) in $\mathrm{Runs}^{\mathcal{A}}_{w}$.

> To (u,q), associate the length of the longest contiguous sequence of green transitions that reach (u,q).

For example:



The red numbers indicate the quantity.

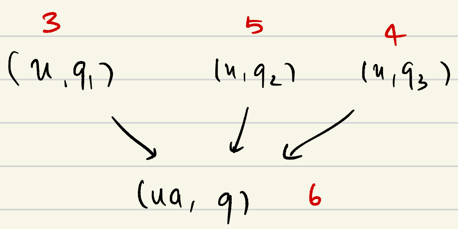
formally:

→ set $\lambda(\epsilon, q_0) = 0$

→ $\lambda(ua, q') = 1 + \max\{\lambda(u,q) \mid (u,q) \text{ is a node } \& \ q \xrightarrow{a:2} q' \text{ in } \mathcal{A}\}$

a green transition (of colour 2)

for eg:

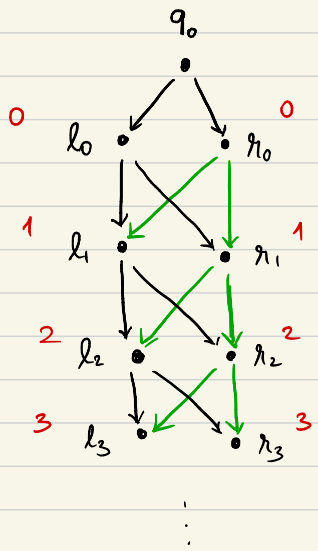


Attempt at Eve's strategy: Consider a strategy where Eve extends her run by choosing successors with a larger λ - in case of tie she chooses arbitrarily.

For example, in the DAG above, Eve plays $q_0 \xrightarrow{a} q_1 \xrightarrow{b} q_2 \xrightarrow{b} q_2 \ldots$

Here it turns out to be an accepting run. Does this strategy always work?

Consider the following DAG:



$$\lambda(l_i) = \lambda(r_i) = i$$

$$\forall i \geq 0$$

By the strategy said earlier, Eve can arbitrarily resolve ties. Hence a possible run would be:

$$q_0 \longrightarrow l_0 \longrightarrow l_1 \longrightarrow l_2 \longrightarrow \cdots$$

However this is a rejecting run.

Therefore we $\boxed{\text{need a more sophisticated strategy!}}$

Eve's strategy in the letter game – the correct one!

Here is where we make use of the results we discussed in the previous lecture.

Recall:

$\mathcal{A}^{safe}$: safety automaton obtained by redirecting all 1-transitions to a rejecting sink state.

Furthermore: for each state q in $\mathcal{A}$, there exists state $p \in WCR(\mathcal{A}, q)$ s.t. Eve wins $G1^{safe}(p; q)$ and $G1^{safe}(p; p)$

Safe-deterministic states: A state p of $\mathcal{A}$ is said to be safe-deterministic if Eve wins $G1^{safe}(p; p)$

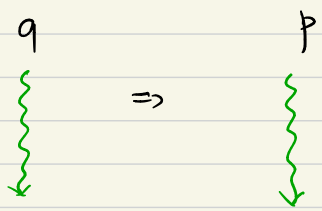
Properties of safe-deterministic states:

Suppose state q is associated to safe-deterministic state 'p'.

→ 1) since Eve wins $G1^{safe}(p; q)$

→ 'green' runs from q can be simulated as a green run from p.

→ $\therefore \mathcal{L}(\mathcal{A}_{safe}, q) \subseteq \mathcal{L}(\mathcal{A}_{safe}, p)$.



A second very important property comes in the next page.

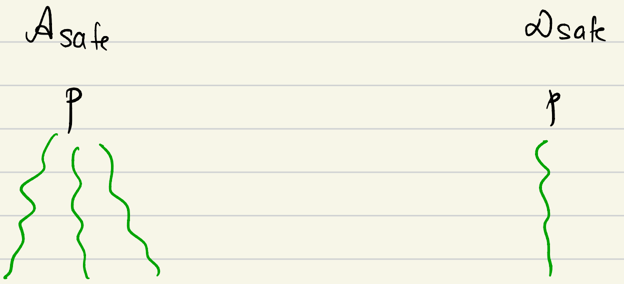
→ 2) There is a deterministic safety automaton $\mathcal{D}_{safe}$ whose states consist of all the safe-deterministic states of $\mathcal{A}_{safe}$ and $\forall$ safe-deterministic states p:

$$\mathcal{L}(\mathcal{D}_{safe}, p) = \mathcal{L}(\mathcal{A}_{safe}, p)$$

The above property can be derived from the following result.

> **Theorem** (Boker, Kupferman, Skrzypczak, FSTTCS'17); (Boker, Lehtinen, LMCS'23):
>
> Let $\mathcal{A}$ be a reachability or safety automaton on which Eve wins the 1-token game.
>
> Then:
>
> - 1. $\mathcal{A}$ is HD.
> - 2. A positional winning strategy in the HD game on $\mathcal{A}$ can be computed in PTIME.
> - 3. $\mathcal{A}$ has a language-equivalent deterministic subautomaton $\mathcal{D}$, which can be constructed in PTIME.

∴ $\mathcal{D}_{safe}$ gives a unique run for all words in the language of safe-deterministic states in $\mathcal{A}_{safe}$.



Now comes a different DAG!

For an infinite word $w \in \Sigma^\omega$ we build a DAG using safe-deterministic states and the automaton $\mathcal{D}_{safe}$.

> Given a word $u \in \Sigma^*$, a state p is said to be **active on u** if
>
> - p is safe-deterministic
> - $p \in WCR(\mathcal{A}, u)$
>
> i.e., there is a run $q_0 \overset{u}{\rightsquigarrow} q$ in $\mathcal{A}$ and $p \in WCR(\mathcal{A}, q)$.

DAG D_w:

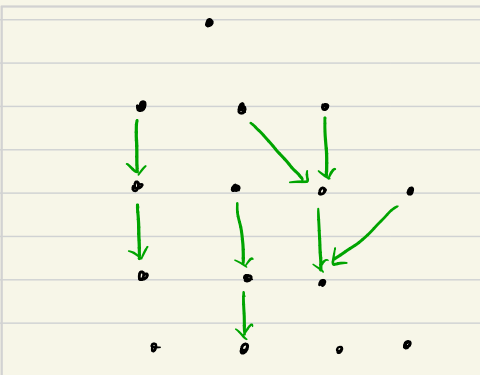
→ Nodes are (u, q) s.t.

- u is a prefix of w
- q is active on u.

→ $(u, q) \to (ua, q')$ if $q \xrightarrow{a} q'$ in $\mathcal{D}_{safe}$ and q' is not sink.

Therefore there is atmost one outgoing edge from every node.

$\mathcal{D}_w$ looks like this:



→ can have nodes without outgoing edges

→ Node may not have an edge coming from an upper level. So it appears new at a level.

Claim: $\mathcal{D}_w$ has an infinite path

Proof: Since $w \in \mathcal{L}(\mathcal{A})$ there is a run which eventually visits only 2-transitions.

Say:

$$q_0 \overset{u}{\rightsquigarrow} q \overset{w'}{\rightsquigarrow} \cdots$$

$$\therefore \quad w' \in \mathcal{L}(\mathcal{A}_{safe}, q)$$
$$= \mathcal{L}(\mathcal{D}_{safe}, q)$$

Furthermore, $\exists$ safe-deterministic state p s.t. $p \in WCR(\mathcal{A}, q)$ and Eve wins $G1^{safe}(p; q)$

Hence:
- 1. $w' \in \mathcal{L}(\mathcal{D}_{safe}, p)$
- 2. (u, p) is a node in $\mathcal{D}_w$.
- 3. There is an infinite path starting from (u, p). This is the unique run in $\mathcal{D}_{safe}$ on w' starting from p.

Now comes Eve's strategy:

- Eve's strategy has access to $\mathcal{D}_{safe}$.
- Eve builds the DAG D_w on the fly as Adam produces the word.
- Suppose Adam produces $w \in \mathcal{L}(\mathcal{A})$.

On a prefix,

→ Eve has produced a run ending in q_u (say)

→ has in its memory a safe-deterministic state r_u that is active on u.

→ Suppose Adam gives a new letter a.

→ Eve extends $q_u \xrightarrow{a} q'$ using strategy

$$G1^{\mathcal{A}}(q_u ; r_u)$$ → as q_u, r_u are weakly co-reachable.

→ If $r_u \xrightarrow{a} r_{ua}$ is a transition in $\mathcal{D}_{safe}$ s.t. r_{ua} is not sink,

then Eve updates memory to r_{ua}.

Else Eve resets its memory to a state r'_{ua} s.t.

in $\mathcal{D}_w$, among all nodes (ua, r), the path

ending at (ua, r'_{ua}) has the longest length

(exercise!)

Claim: When $w \in \mathcal{L}(\mathcal{A})$, after a point there are no 'resets'.

Eve produces a run r_u ⟿ ... in its memory

Suppose Eve was at q_u when memory is at r_u. Since

Eve plays according to $G1^{\mathcal{A}}(q_u ; r_u)$, the run produced

from q_u is also accepting!

Hence, this strategy produces an accepting run in Eve's token

→ This proves $\mathcal{A}$ is HD!