TIMED AUTOMATA

LECTURE 21

Goals for today:

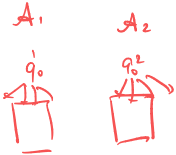
Alternating Timed Automata

↳ closure properties

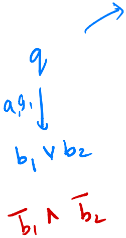
→ 1- clock restriction

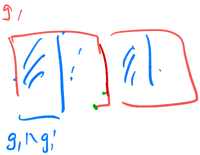
→ Expressive power of 1- ATA

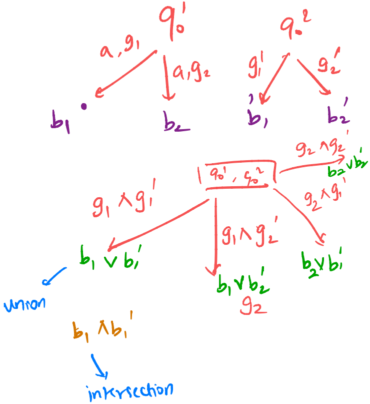
Closure properties



- Union, intersection: use disjunction/conjunction
- Complementation: **interchange**
  1. acc./non-acc.
  2. conjunction/disjunction







Complementation:

$$q \xrightarrow{a,\, g} [(q_1, r_1) \wedge (q_2, r_2)] \vee (q_3, r_3) \vee [(q_4, r_4) \wedge (q_5, r_5)]$$

$\downarrow$ complementation:

$$q \xrightarrow{a,\, g} [(q_1, r_1) \vee (q_2, r_2)] \wedge (q_3, r_3) \wedge [(q_4, r_4) \vee (q_5, r_5)]$$

Closure properties

- Union, intersection: use disjunction/conjunction
- Complementation: **interchange**
  1. acc./non-acc.
  2. conjunction/disjunction

No change in the number of clocks!

Section 2:

The 1-clock restriction

- Emptiness: given A, is $\mathcal{L}(A)$ empty
- Universality: given A, does $\mathcal{L}(A)$ contain all timed words
- Inclusion: given A, B, is $\mathcal{L}(A) \subseteq \mathcal{L}(B)$

- Emptiness: given A, is $\mathcal{L}(A)$ empty
- Universality: given A, does $\mathcal{L}(A)$ contain all timed words
- Inclusion: given A, B, is $\mathcal{L}(A) \subseteq \mathcal{L}(B)$

Undecidable for **two clocks or more** (~~via Lecture 3~~)

Universality problem for NTA $\longrightarrow$ Emptiness problem for ATA

$\mathcal{B}$ $\longrightarrow$ Look at $\mathcal{B}$ as an ATA.

Complement it and check for emptiness

$\mathcal{L}(\mathcal{B})$ is universal iff

$\overline{\mathcal{L}(\mathcal{B})}$ is empty

- Emptiness: given A, is $\mathcal{L}(A)$ empty
- Universality: given A, does $\mathcal{L}(A)$ contain all timed words
- Inclusion: given A, B, is $\mathcal{L}(A) \subseteq \mathcal{L}(B)$

Undecidable for **two clocks or more** (~~via Lecture 3~~)

Decidable for **one clock** (~~via Lecture 4~~)

- Emptiness: given A, is $\mathcal{L}(A)$ empty
- Universality: given A, does $\mathcal{L}(A)$ contain all timed words
- Inclusion: given A, B, is $\mathcal{L}(A) \subseteq \mathcal{L}(B)$

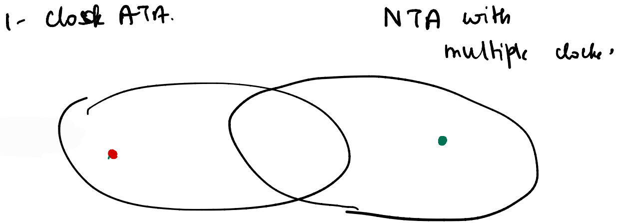
Undecidable for **two clocks or more** (~~via Lecture 3~~)

Decidable for **one clock** (~~via Lecture 4~~)

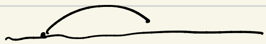
Restrict to one-clock ATA

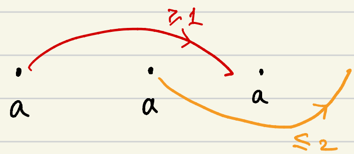
Theorem

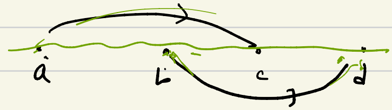
Languages recognizable by 1-clock ATA and (many clock) TA are **incomparable**



Alternation vs. Multiple clocks:

Alternation:  For every point, $\exists$ another at distance c.

Multiple clocks: 

Interleaving. 

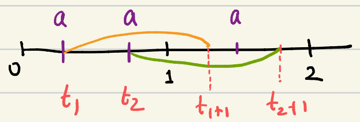
Need multiple clocks

Example L: no a's at distance 1. $\rightarrow$ 1-clock ATA.
but no NTA.

Example of a language accepted using multiple clock T.A.,
but not 1-ATA?

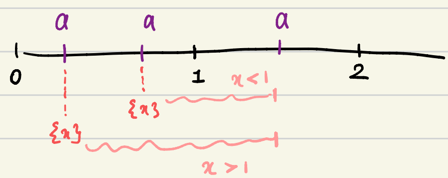
Clarification about the expressive power of 1-ATA:

Question: Let $L_1 = \{ (aaa, t_1 t_2 t_3) \mid 0 < t_1 < t_2 < 1,\ t_1 + 1 < t_3 < t_2 + 1 \}$



Can you construct a 1-ATA for L_1?

Idea:



1-ATA:

$(q_0, a, 0<x<1) \longmapsto (p_1, \{x\}) \wedge (s_1, \phi)$

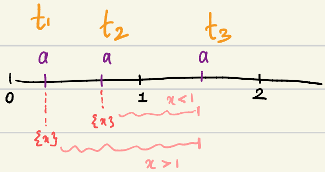
$x>0$ $\quad (p_1, a, \cancel{\text{true}}) \longmapsto (p_2, \phi)$

$(s_1, a, x<1) \longmapsto (s_2, \{x\})$

$(p_2, a, x>1) \longmapsto (f, \phi)$

$(s_2, a, x<1) \longmapsto (f, \phi)$

$(f, a, \text{true}) \longmapsto (\text{reject}, \phi)$, $\quad (\text{reject}, a, \text{true}) \longmapsto (\text{reject}, \phi)$



<u>1-ATA:</u> $(q_0, a, 0<x<1) \longmapsto (p_1, \{x\}) \wedge (s_1, \phi)$

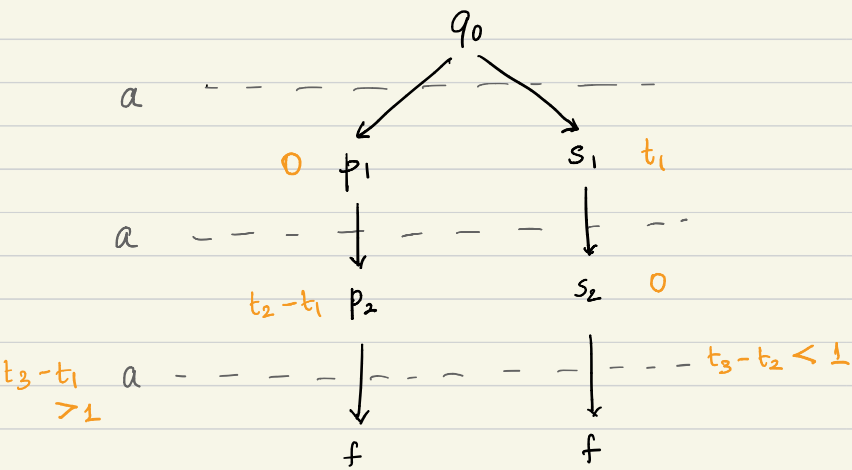
$(p_1, a, \overset{x>0}{\cancel{true}}) \longmapsto (p_2, \phi)$

$(s_1, a, x<1) \longmapsto (s_2, \{x\})$

$(p_2, a, x>1) \longmapsto (f, \phi)$

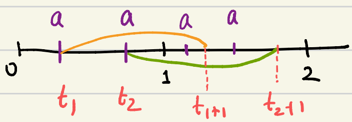
$(s_2, a, x<1) \longmapsto (f, \phi)$

$(f, a, true) \longmapsto (reject, \phi)$, $(reject, a, true) \longmapsto (reject, \phi)$

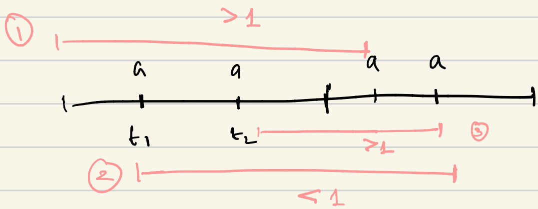


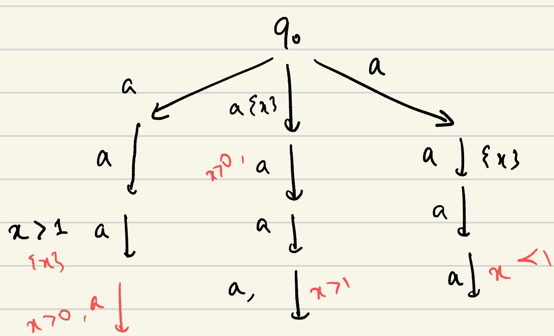
A small modification of the previous example:

Question: Let $L_2 = \{ (aaaa, t_1 t_2 t_3 t_4) \mid 0 < t_1 < t_2 < 1, \; 1 < t_3 < t_4, \; t_1 + 1 < t_4 < t_2 + 1 \}$



Can you construct a 1-ATA for L_2?





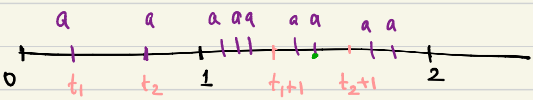
Question:

$$\mathcal{L}_3 = \{ (a^k, t_1 t_2 \ldots t_k) \mid k \geq 3 :$$

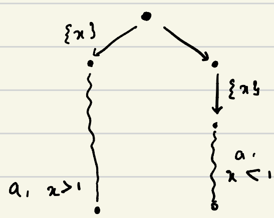
$$0 < t_1 < t_2 < 1$$

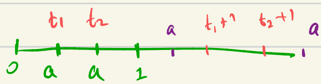
$$\exists j \geq 3 \text{ s.t. } t_1 + 1 < t_j < t_2 + 1 \quad \}$$

$$t_3 > 1$$

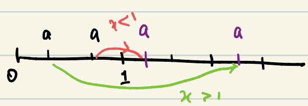


Problem:





The two a's could be different

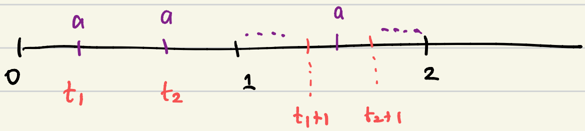


- This is an intuition that $\mathcal{L}_3$ cannot be accepted by a 1-ATA.
- However, proving that a language cannot be accepted by a 1-ATA is difficult.
- We will see another example given in the paper, for which there is a proof that it cannot be accepted by a 1-ATA.

$L = \{ (a^k, t_1 t_2 \dots t_k) \mid 0 < t_1 < t_2 < 1$

$1 < t_3, \dots, t_k < 2$

there is exactly one a between $t_1 + 1$ and $t_2 + 1$ $\}$



- L can be accepted by a deterministic T.A with 2 clocks.

Goal: To prove that L cannot be accepted by a 1-ATA.

Step 1: Understand some property of DFAs

Step 2: How Step 1 translates to untimed alternating finite automata

Step 3: Any 1-ATA accepting L behaves like an untimed AFA in the interval $(1, 2)$, where clocks are useless.

Step 4: Use Step 1 and 2 in 3 to get a contradiction.

Step 1: Understanding a property of DFAs.

- Consider a unary alphabet $\{a\}$, and DFA $\mathcal{B} = (Q, q_0, \delta, F)$
- For each a^k, the DFA gives rise to a function

$$f_k^{\mathcal{B}} : Q \longmapsto Q$$

$$\boxed{\begin{array}{lcl} q_1 & \xrightarrow{a^k} & q_1' \\ q_2 & & \\ \vdots & & \\ q_n & \longrightarrow & q_n' \end{array}}$$

- The number of functions from $Q \longmapsto Q$ is finite.
- Therefore, if we look at the sequence:

$$f_1^{\mathcal{B}},\ f_2^{\mathcal{B}},\ f_3^{\mathcal{B}},\ \ldots\ldots.$$

there exist m, ℓ, s.t.

$$\boxed{f_m^{\mathcal{B}} = f_{m+\ell}^{\mathcal{B}}}$$

m $\quad$ $m+\ell$

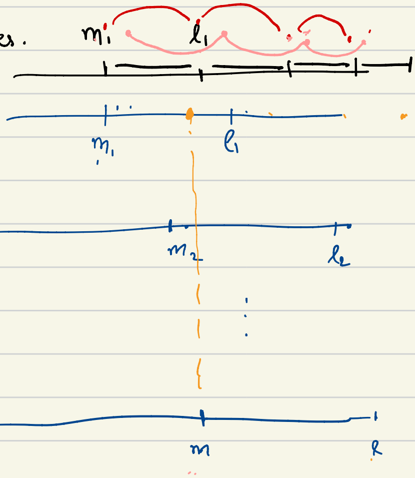
f_m: $q_1 - q_1'$, $q_2 - q_2'$

$f_{m+\ell}$: $q_1 - q_1'$, $q_2 - q_2'$

f_{m+1}: $q_1 \longrightarrow \delta(q_1', a)$, $q_2 - \delta(q_2', a)$ $\quad = f_{m+\ell+1}$

- Moreover: $\quad f_{m+i}^{\mathcal{B}} = f_{m+\ell+i}^{\mathcal{B}} \quad \forall i \geq 0$

Consider all DFA with atmost n states.



finitely many

$$\left\{\begin{array}{lll} \mathcal{B}_1 & - & m_1, l_1 \\ \mathcal{B}_2 & - & m_2, l_2 \\ \vdots & \vdots & \\ \mathcal{B}_j & - & m_j, l_j \end{array}\right.$$

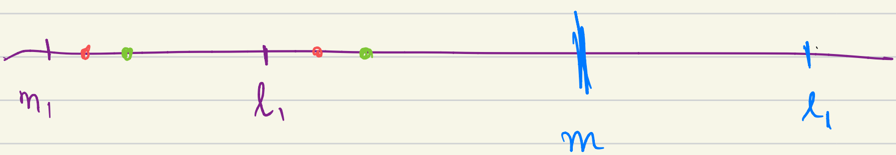
- Let $m = \max(m_1, \ldots m_j)$

$l = l_1 \cdot l_2 \cdot l_3 \cdots l_j$

Then for every DFA $\mathcal{B}$ with $\leq n$ states, we have:

$$f^{\mathcal{B}}_{m+i} = f^{\mathcal{B}}_{m+l+i} \qquad \forall i \geq 0$$

$f_{m+i} = f_{m+kl_1+i}$



$$f_{m_1+i} = f_{m_1+l_1+i} = f_{m_1+2l_1+i} = f_{m_1+3l_1+i}$$