# GOALS:

- Event-Clock Automata $\longrightarrow$ Timed Automata
- Expressiveness of various models

## ECA to NTA:

### Part 1: ERA to timed automata

$$\cdot \xrightarrow[\{x_a\}]{a}$$

add reset of $x_a$ to all transitions having $a$

$$q \xrightarrow{a,\ x_b \sim c} q' \quad \longrightarrow \quad q \xrightarrow[\{x_a\}]{a,\ x_b \sim c} q'$$

in ECA — in T.A.

$$q \xrightarrow{a,\ x_b = \perp} q' \quad \longrightarrow \quad ??$$

in ECA

- To simulate guards having $\perp$, we maintain information in the state.

$$\boxed{\begin{array}{c} q \\ x_a = \perp \\ x_b = \perp \end{array}} \xrightarrow{a} \boxed{\begin{array}{c} q' \\ \cdot \\ x_b = \perp \end{array}}$$

## ERA to T.A:

Given ERA $\mathcal{A} = (Q, \Sigma, Q_0, \Delta, F)$

T.A equivalent to $\mathcal{A}$ is given by:

**States:** $(q, \psi)$ where $q \in Q$

$\psi \subseteq \{ x_a = \perp \mid a \in \Sigma \}$

**Initial state:** $(q_0, \psi_0)$ s.t. $q_0 \in Q_0$

$\psi_0 := \{ x_a = \perp \mid a \in \Sigma \}$

**Accepting:** $(q, \psi)$ s.t. $q \in F$

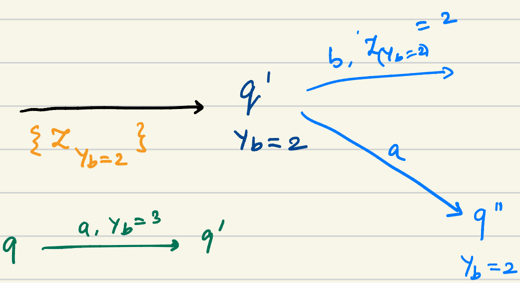
**Transitions:** $(q, \psi) \xrightarrow[R]{a,\, g} (q', \psi')$ if

$q \xrightarrow{a,\, \varphi} q'$ in ERA

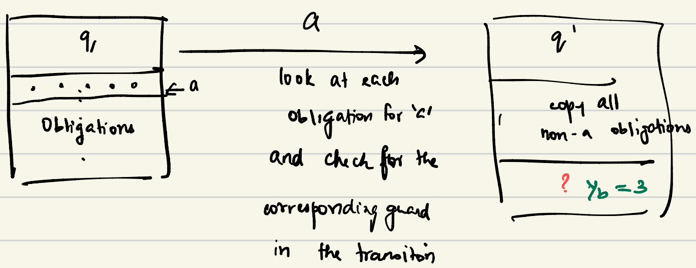
- $(q, a, \varphi, q') \in \Delta$
- if $\varphi$ contains $x_b = \perp$ for some $b \in \Sigma$, then $(x_b = \perp) \in \psi$
- $g$ is conjunction of all constraints $x_a \sim c$ in $\varphi$
- $\psi' = \psi \setminus \{ x_a = \perp \}$
- $R$ is $\{ x_a \}$

# Part 2: EPA to NTA

General idea: $\xrightarrow{y_b = 2} q'$




States of T.A: $q \xrightarrow{a,\ y_b=3} q'$

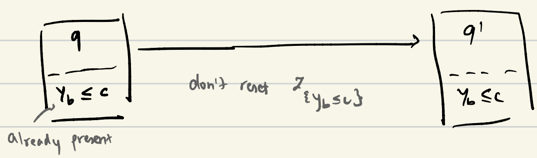


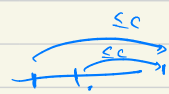

Reset $\{\mathcal{Z}_{y_b=3}\}$

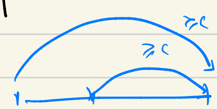
---

* If obligation is $y_b = \perp$, then no 'b' transition should exist from that state.

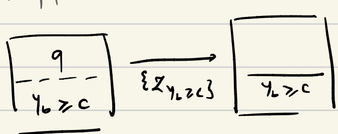
* $q \xrightarrow{y_b \le c} q'$



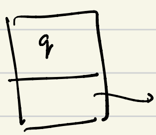




* $q \xrightarrow{y_b \ge c} q'$




* What are accepting state?




$q \in F$ (in the EPA)

only obligations are of the form $y_a = \perp$

## EPA to NTA:

Given EPA $\mathcal{A} = (Q, \Sigma, Q_0, \Delta, F)$.

The equivalent NTA is given as follows:

Let $\Phi_A$ be the set of all atomic clock constraints appearing in edges of A.

Atomic clock constraints: $y_a = \perp$ or $y_a \sim c$

$\sim \in \{<, \leq, >, \geq\}$

- States: $(q, \psi)$ $\xrightarrow{}$ obligations

  $q \in Q$

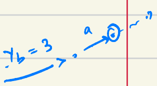
  $\psi \subseteq \Phi_A$

- Initial state: $(q_0, \psi_0)$

  $q_0 \in Q_0$

  $\psi_0$ does not contain a constraint of the form $(y_a \sim c)$

  $\psi_0 \subseteq \{ y_a = \perp \mid a \in \Sigma \}$




- Accepting state: $(q, \psi)$

  $q \in F$

  $\psi \subseteq \{ y_a = \perp \mid a \in \Sigma \}$

- Clocks: For every $\psi := (y_a \sim c) \in \Phi_A$, keep a clock $z_\psi$

- **Transitions:** $(q, \psi) \xrightarrow[R]{a,\ g} (q', \psi')$ if

- i) $(q, a, \varphi, q') \in \Delta$ $\qquad q \xrightarrow{a,\ \varphi} q'$ in EPA
- ii) Constraint $y_a = \perp$ does not appear in $\psi$
- iii) $g$ : Conjunction of $z_{(y_a \sim c)} \sim c$ for every $y_a \sim c \in \psi$
- iv) For all $b \neq a$, if a constraint involving $y_b$ appears in $\psi$, then it appears in $\psi'$ too
- v) Each atomic constraint of $\varphi$ appears in $\psi'$.
- vi) For each 'b', and for $\sim$ equal to $>$ or $\geq$, $z_{y_{b \sim c}}$ appears in reset condition $R$ iff constraint $y_b \sim c$ is present in $\varphi$.
- vii) For each 'b', and for $\sim$ equal to $<$ or $\leq$, $z_{y_{b \sim c}}$ appears in reset condition $R$ iff constraint $y_b \sim c$ is present in $\varphi$ and either $b = a$ or the constraint $y_b \sim c$ is not present in $\psi$.

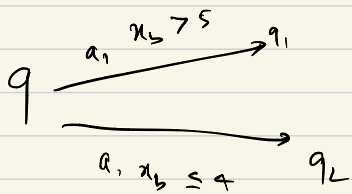
## Part 3: ECA to NTA:

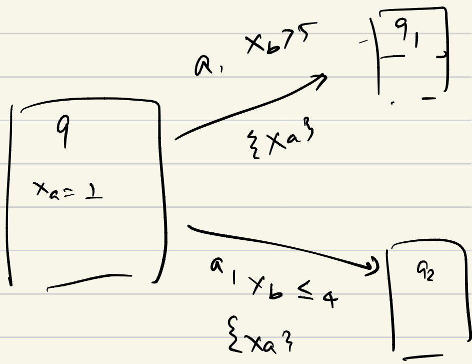
Combination of both methods

**Question:**

- 1. Does ERA to T.A. preserve determinism? Yes.
- 2. Does EPA to T.A. preserve determinism?

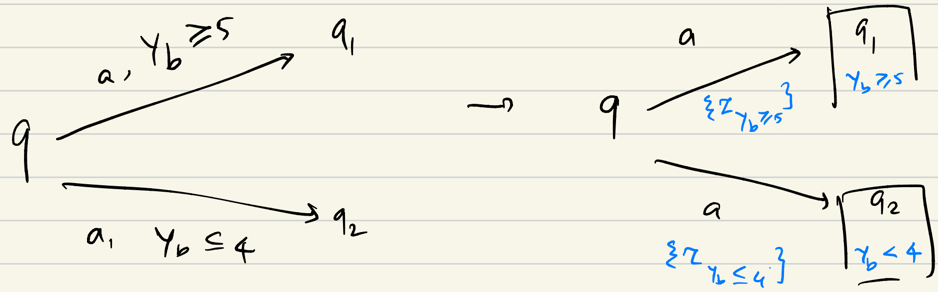
1). ERA to T.A.




- guards are maintained as they are in the ERA.

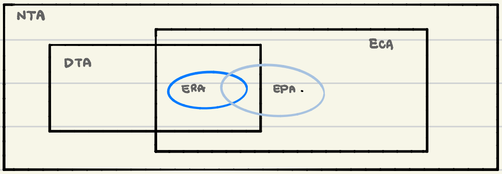
This will ensure that if we start with a DERA, we will get a D.T.A.




Not deterministic

$\therefore$ EPA to T.A. conversion does not preserve determinism.

## Expressive Power of different classes:




1) ERA $\subseteq$ DTA: conversion preserves determinism.

2) ERA $\not\subseteq$ EPA

3) EPA $\not\subseteq$ ERA

$\underbrace{a\ a\ b}_{=1}$ → EPA, but not ERA

$\underbrace{a\ b\ b}_{=1}$ → ERA, but not EPA

4) DTA $\not\subseteq$ EPA ($\exists$ L in DTA which is not EPA recog.)

$\underbrace{abb}_{=1}$

5) EPA $\not\subseteq$ DTA    $\exists$ a L which EPA but not by DTA

$\{ a^k b \mid k \geq 1$, $\exists$ some 'a' which is at distance 1 from 'b'$\}$

(see next page)

5) DTA $\not\subseteq$ ECA ($\exists$ L in DTA which is not ECA recog.)

$\overbrace{a\ a\ a}^{=1}$

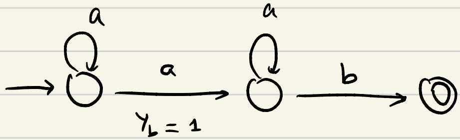
6) ECA $\not\subseteq$ DTA    $\exists$ L in ECA which is not DTA recog.

7) ECA $\not\subseteq$ EPA $\cup$ ERA : $\{a\underbrace{ab}_{\geq 1}\} \cup \{\underbrace{abb}_{\geq 1}\}$

8) ECA $\subseteq$ NTA: conversion algorithms as seen before

EPA $\not\subseteq$ DTA

$\{ a^k b \mid k \geq 1,$ there exists an 'a' which is at distance 1 from 'b' $\}$




There is no DTA for this language. Intuitively, we cannot guess the 'a' deterministically for which the b is at distance 1.

Exercise: Prove this formally.

## Summary:

- Event - clock automata.

- $X_a$ : records time since last 'a' event
- $Y_a$: predicts time to next 'a' event

- Determinized

- Closed under boolean operations.

**Inclusion:** $\mathcal{L}(B) \subseteq \mathcal{L}(A)$

$\mathcal{L}(B)$ ↑ T.A ; $\mathcal{L}(A)$ ↑ E.C.A

↓

Let $A^c$ be the ECA for $\mathcal{L}(A)$

↓

$D$ be the NTA equivalent to $A^c$

$$\mathcal{L}(B) \subseteq \mathcal{L}(A) \iff \mathcal{L}(B) \cap \mathcal{L}(A^c) = \phi$$
$$\iff \mathcal{L}(B) \cap \mathcal{L}(D) = \phi$$

↳ N.T.A.

- Decidable if $A$ is an ECA.

- Expressive power of the model.