TIMED AUTOMATA

LECTURE 21

GOALS OF TODAY'S LECTURE

- Updatable Timed automata
  - Introduction to the model

Reference:

- Updatable Timed Automata
  Bouyer, Dufourd, Fleury, Petit
  Theoretical Computer Science (2004)

Updatable Timed automaton (UTA):

Resets generalized to **updates**

Updates:

X: a set of clocks

For each clock $x \in X$, an **update on x** takes the following form:

$$x := c \quad | \quad x := y + d \qquad c \in \mathbb{N},\ d \in \mathbb{Z},\ y \in X$$

Examples: $x := 5$, $x := x - 1$, $x := y + 2$, $x := y$, $x := 2 - 1$

An **update function** associates to each clock $x \in X$, an update of the above form.

Denote the set of all update functions as $\mathcal{U}(X)$

Remark: For notational simplicity, we refer to each update function in $\mathcal{U}(X)$ as an update.

UTA : Syntax:

$$(Q, Q_0, \Sigma, X, \Delta, F)$$

- Q: States
- Q_0: initial states
- Σ: alphabet
- X: clocks
- Δ: Transitions
- F: Final states

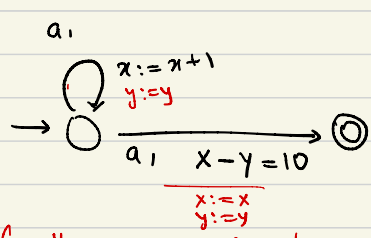
Transition relation Δ:

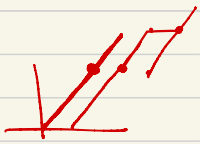
$$q \xrightarrow[up]{a,\ g} q'$$

$$\Delta \subseteq Q \times \Sigma \times Guards(X) \times \mathcal{U}(X) \times Q$$

- $Guards(X)$: includes diagonal and diagonal-free constraints
- $\mathcal{U}(X)$: an update

Example:



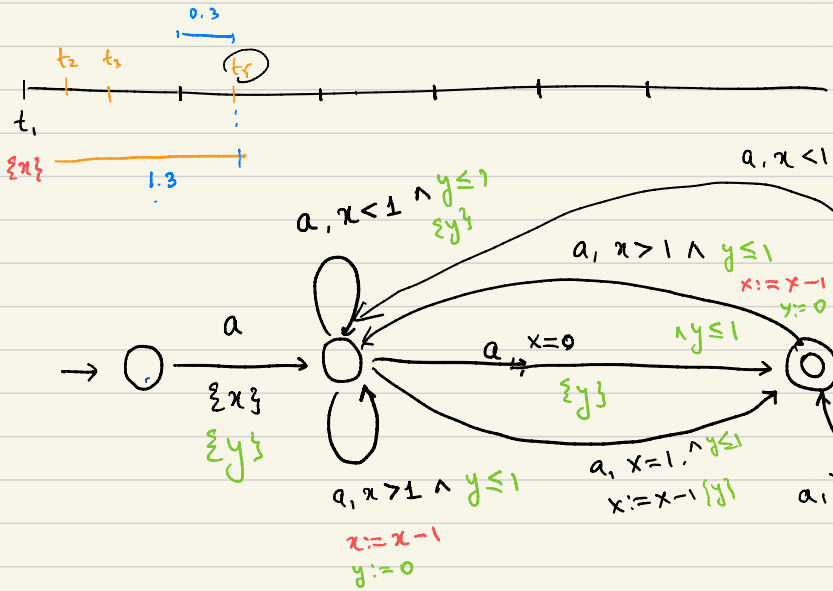


$$\{ (a^n, t_1 t_2 \ldots t_n) \mid t_1 \leq t_2 \ldots \leq t_n \}$$

Problem 1:

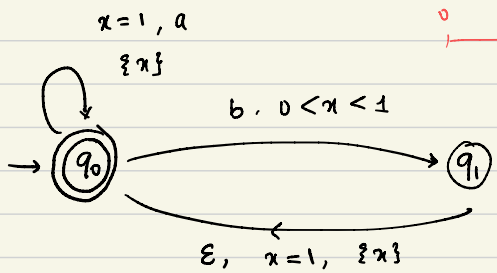
Construct a UTA for the following language:

$$\{ (a^n, t_1 t_2 \dots t_n) \mid n \geq 2,\ t_n - t_1 \in \mathbb{N},\ t_{i+1} - t_i \leq 1\ \forall i < n \}$$



Problem 2:

1) What is the language of the following timed automaton with ϵ-transitions?



2) Build a UTA (without ϵ-transitions) whose language is equivalent to the above automaton.

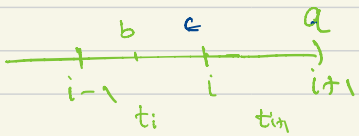
---

1)

Language of above automaton: $(a_1 a_2 a_3 \dots a_n, t_1 t_2 \dots t_n)$

$i \geq 1$

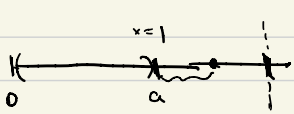
$$\begin{cases} a_i = a & \text{if } t_i = i \\ a_i = b & \text{if } i-1 < t_i < i \end{cases}$$

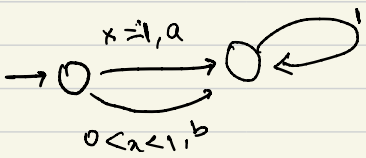


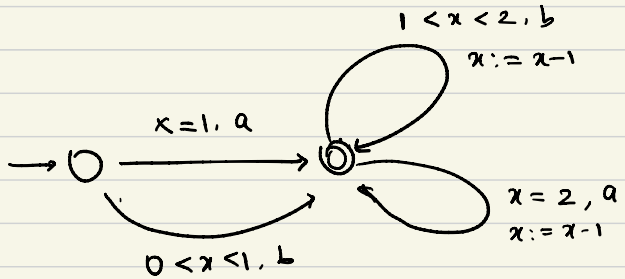
$t_i \in (i-1, i]$

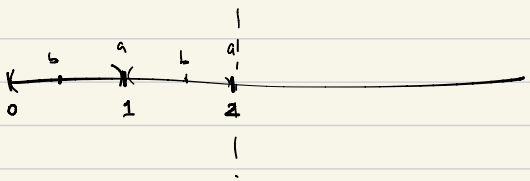
---

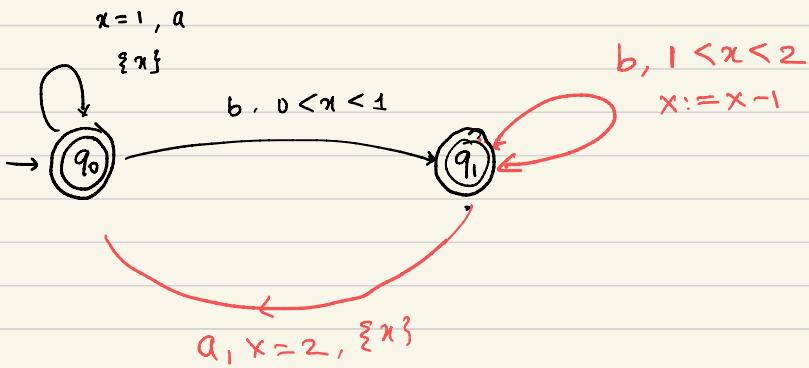
2)






1 < x < 2, b
x := x − 1
x = 1, a
x = 2, a
x := x − 1
0 < x < 1, b


b
a
b
a
0
1
2


x = 1, a
{x}
b, 1 < x < 2
x := x − 1
b, 0 < x < 1
q0
q1
a, x = 2, {x}

Summary:

- Introduction to UTA

Questions on UTA:

- Expressive power of UTA compared to TA, TA + ε
- Emptiness problem for UTA