Timed Automata: Automates Temporisés

Cours TOV

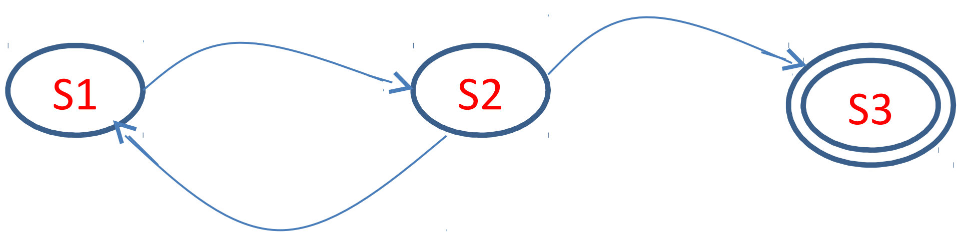
M1: GLSD

L.Kahloul

Timed Automata why? (1)

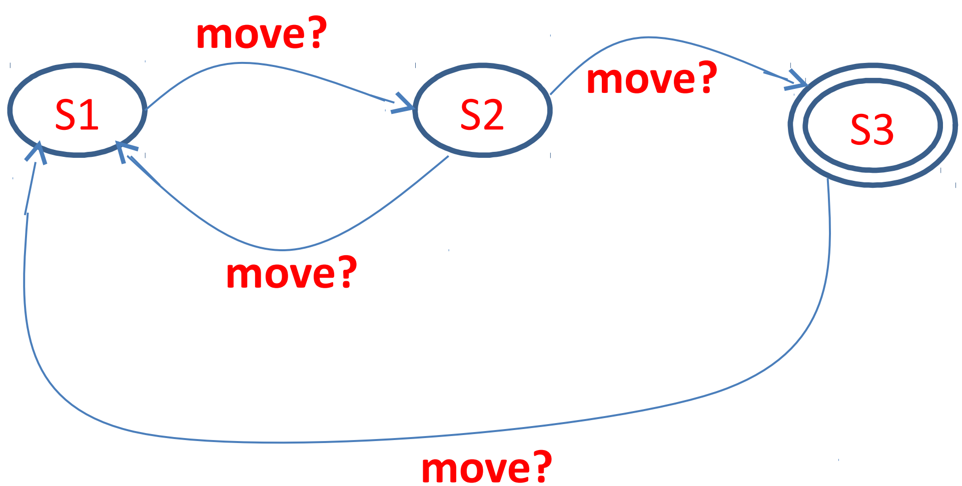
- To give more **expressiveness** for models;
- Example:

  Modelling a train which serves three stations S1, S2, S3. The train can move from S1 to S2, from S2 to S3, from S2 to S1, and from S3 to S1:



Timed automata why? (2)

- Then: in order to move from station to station, the train **<u>needs to receive a command</u>**: "move". The model must be as following.



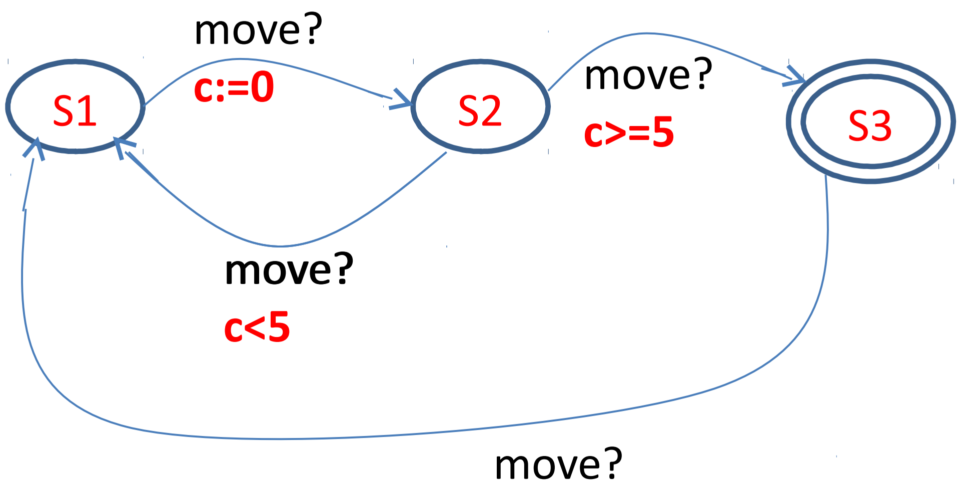
- **move?** Is **<u>an action</u>** and **move!** Is called its **<u>co-action</u>** (and vice versa)

Timed automata why? (3)

- Next: in order to move from station to station the train uses a clock c.

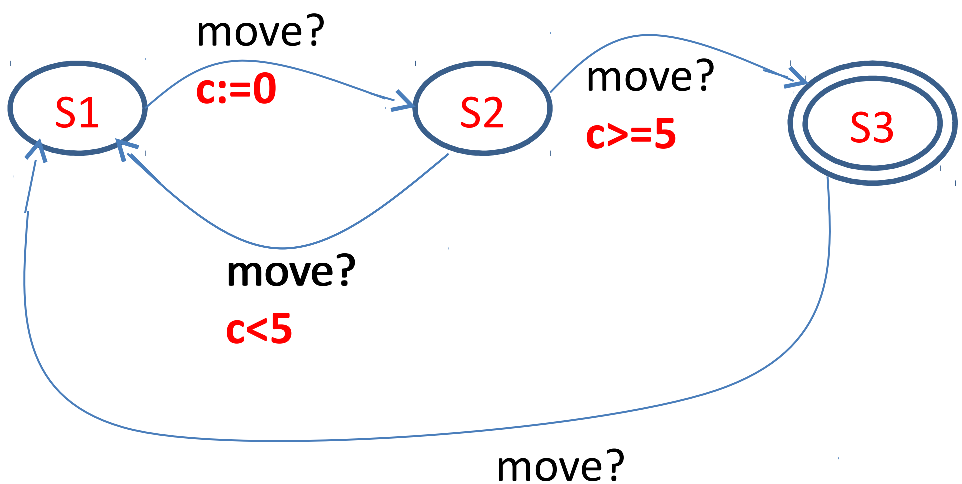
- If the train is in the s1 and it receives the command "move", it resets c and moves to S2. In S2,
- if the train receives the command before 5 seconds then it will return to S1 else, it will go to S3.
- In S3, the train waits the command to return to S1.

Timed automata why? (4)



- C:=0 is called a **<u>reset action</u>**;
- C<5, c>=5 are called **<u>guards</u>**. They are the conditions to be fulfilled to transit the edge;

Timed automata
why?

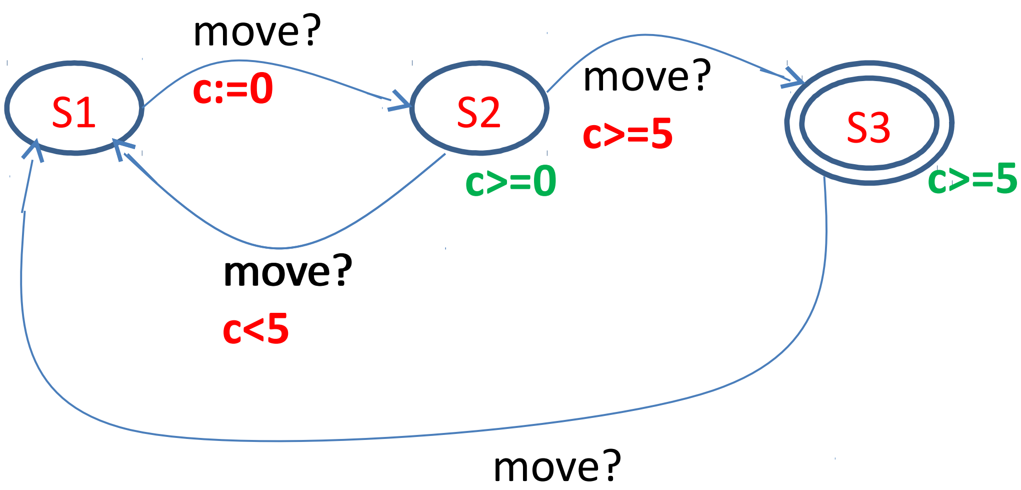


- Finally:

-On the stations S2, we have always c>=0,

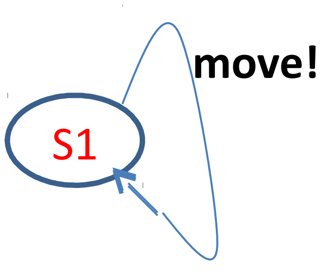
-On the station S3, we have always c>=5.

These two logic expressions are called **<u>invariants</u>**.



Timed automata why?

- The previous automaton can be, now, synchronized with the following one:



Timed automata
Definition

- A timed automaton (informally):
  - a **finite-state** machine;
  - extended with **clock variables (real values);**
  - All the clocks **progress synchronously**.

Timed automata:
Formal Definition

A timed automaton is a tuple:

TA=(***L*, *l_0*, *C***, ***A*, *E*, *Inv***), suh that:

- ***L*** is a set of locations,
- ***l_0*** $\in$ ***L*** is the initial location,
- ***C*** is the set of clocks,

Timed automata:

Formal Definition

TA=(***L, l₀, C, A, E, Inv***),

- ***A*** is a set of **actions**, **co-actions** and the internal **τ-action**,
- $\boldsymbol{E \subseteq L \times A \times B(C) \times 2^C \times L}$ is a set of edges between **locations** with an **action**, a **guard** and a **set of clocks to be reset**,
- $\boldsymbol{Inv : L \rightarrow B(C)}$ assigns **invariants** to locations.

Timed automata:
Formal Definition

Remark:

B(C) is the set of **conjunctions** over simple conditions of the form $c \; \alpha \; n$ or $c_1 - c_2 \; \alpha \; n$, where:

- $c, c_1, c_2 \in C$ (i.e. c, c_1 and c_2 are clocks)
- $n \in N$ (n is a natural number)
- $\alpha \in \{<, \leq, =, \geq, >\}$

Timed automata:

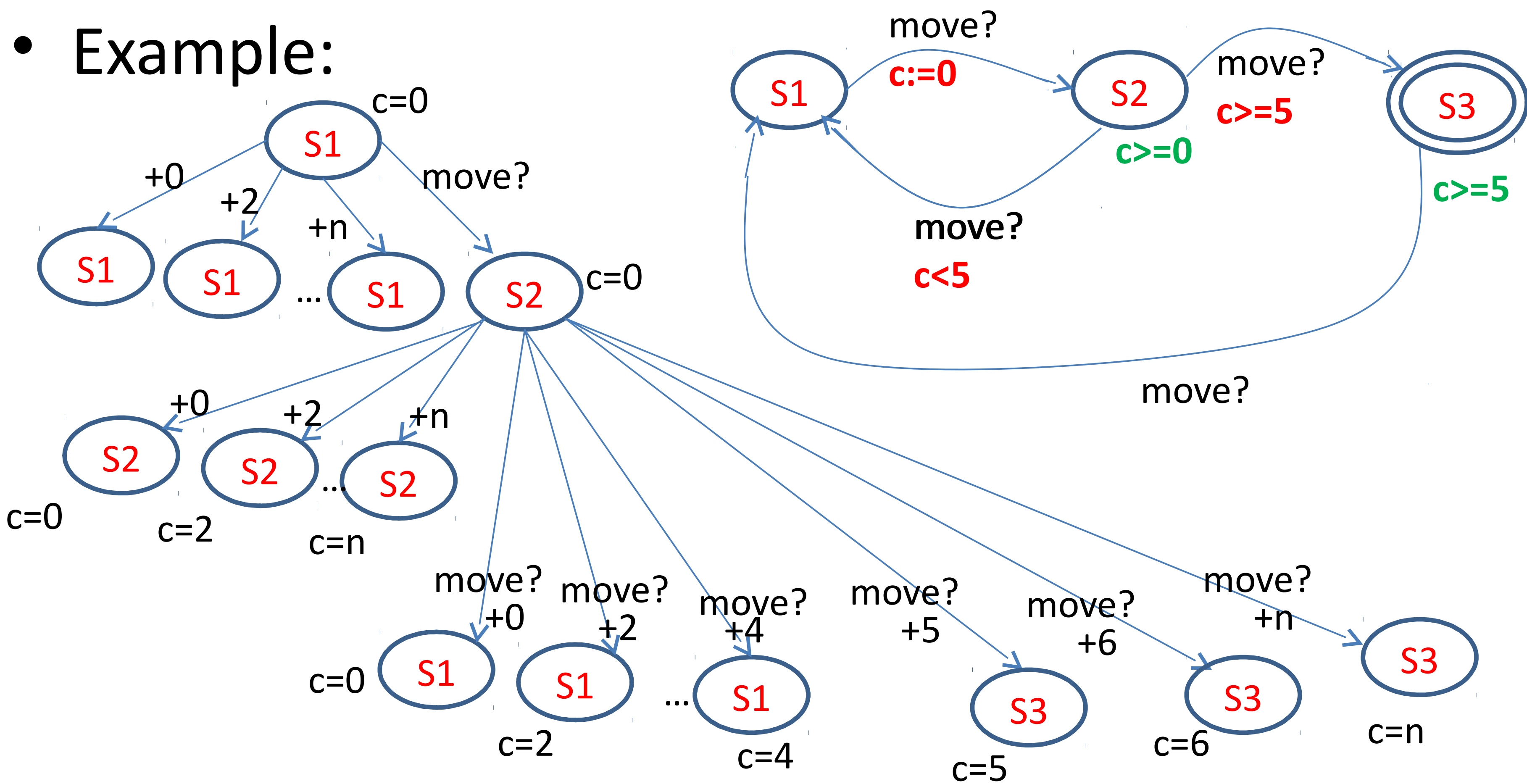
Semantics: How the automaton is executed?

- The semantics of a timed automaton is given through an **LTS** (**Labelled Transitions System**),
- An **LTS** is an **infinite automaton** and it represents the execution of the Timed Automaton,
- Each state of the LTS represents a "**state of the TA**" with a "**valuation of the set of clocks**" (i. e: values of the clocks at this state),
- The "in**itial state**" of the LTS represents the "**initial location**" of the TA with the "**initialisation"** of the set of clocks,
- Each edge in the LTS will be **labelled**

Timed Automata:

Semantics: How the automaton is executed?

- Example:



Timed automata:

Semantics: How the automaton is executed?

- The execution of the automaton means to update the values of the clocks ➔ clocks will have values by a function ***u***
- The function ***u*** (the valuation of clocks) is defined as: $\boldsymbol{u : C \rightarrow R_{\geq 0}}$
- We have:

  $\boldsymbol{u_0(c)=0}$ for each clock ***c*** in ***C***

  $\boldsymbol{R^C}$ is the set of all clock valuations

Timed automata:

Formal Semantics

If TA=(***L, l_0, C, A, E, Inv***) is a timed automaton, then its LTS is the triple: (***S, s_0,***$\rightarrow$), where:

- ***S*** $\subseteq$ ***L x R^C*** (each state in the LTS is a location in te AT with the valuation of clocks at this location)
- ***s_0=(l_0, u_0)*** (the initial state in the the LTS is the intial location in te AT with the initialisation of all the clocks)
- $\rightarrow \subseteq S\times(R_{\geq 0}\cup A)\times S$ (a triple composed of a source state, a destination state, and a label of the edge. The label is a couple: values of clocks and an action)

Timed automata:

Formal Semantics

The transition relation → is defined for each clock ***x*** as:

(1) A delay transition: The clock ***x*** changes in the same location:

$$(l, u(x)) \rightarrow^{d} (l, u(x)+d)$$

if ∀d': $0 \leq d' \leq d \Rightarrow u(x)+d' \in Inv(l)$,

Timed automata:
Formal Semantics

(2) **An action transition**: The location changes from ***l*** to ***l' :***

$$(l, u(x)) \rightarrow^{a} (l', u'(x))$$

if there exists an edge $e=(l, a, g, r, l')\in E$

(***l*** for source location, ***a*** for action, ***g*** for guard, ***r*** for clocks to be reset, and ***l'*** for destination location)

such that:

- $u(x)\in g$: means $u(x)$ satisfies the guard g
- $u'(x)\in Inv(l')$: means $u'(x)$ satisifes the invariant in l'
- $u'(x)=u_0(x)$, resets the clock x to be reset (x in r)

Network of Timed Automata

- The real systems require the use of several synchronised automata;

- These automata **share** a set of actions and clocks;

- How the **semantics** of a network of automata will be defined?

Network of Timed Automata

« composition »

Let A1=($\boldsymbol{L_1, l_{01}, C_1, A_1, E_1, Inv_1}$) et A2= ($\boldsymbol{L_2, l_{02}, C_2, A_2, E_2, Inv_2}$). The composition of A1 and A2 is the TA A3=($\boldsymbol{L_1 x L_2, (l_{01}, l_{02}), C_1 \cup C_2, A_1 \cup A_2, E, Inv}$), such that:

- ***E***: for each two edges: ***e1=(l1, a1, g1, r1, l'1) , e2=(l2, a2, g2, r2, l'2),*** we have two cases:

- Progression in one automaton

$$e=((l1,l2), a1, g1, r1, (l'1,l2))$$

or

$$e=((l1,l2), a2, g2, r2, (l1,l'2))$$

- Synchronisation

$$e=((l1,l2), a1a2, g1 \wedge g2, r1 \cup r2, (l'1,l'2))$$

- For all ***(l, l')*** in $\boldsymbol{L_1 x L_2}$: ***Inv(l,l')=Inv(l)∧Inv(l')***

Networks of Timed Automata

« Semantics: Formally»

- Let $A_i = (L_i, l^0_i, C, A, E_i, I_i)$ be a network of n timed automata. Let $l_0 = (l_{01}, \ldots, l_{0n})$ be the initial location vector.

- The semantics is defined as a transition system $(S, s_0, \rightarrow)$, where:

  - $S = (L_1 \times \ldots \times L_n) \times R^C$ is the set of states,
  - $s_0 = (l_0, u_0)$ is the initial state,
  - $\rightarrow \subseteq S \times S$ is the transition relation defined by:

Networks of Timed Automata

« Semantics: informally»

Informally, Three cases are possible

1) A **delay transition**: where the network does not change location. Only **a progression in the clocks**. This progression must satisfy the invariants of the location;

2) A **silent transition**: **one location** is changed in the vector of locations. The silent transition must reset the necessary clocks;

3) A **synchronisation transition**: **two locations** will be updated in the vector. This change must respect the invariant of the new location.

Networks of Timed Automata

« Semantics: Formally»

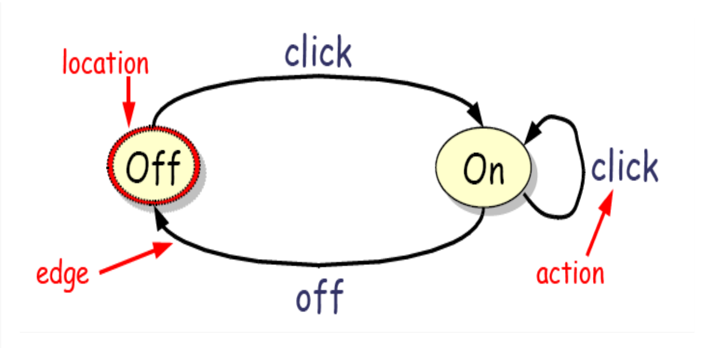
-delay transition: $(l\ ,u) \xrightarrow{d} (l, u+d)$ *if* $\forall d' : 0 \le d' \le d \Rightarrow u + d' \in Inv(l)$.

- Silent Action transition: $(l\ ,u) \xrightarrow{\tau} (l[l'_i\ /\ l_i], u')$ *if there exists* $l_i \xrightarrow{g,\tau,r} l'_i$ *such that* $u \in g$, $u' = [r \rightarrow 0]u$ *and* $u' \in Inv(l[l'_i\ /l_i])$.

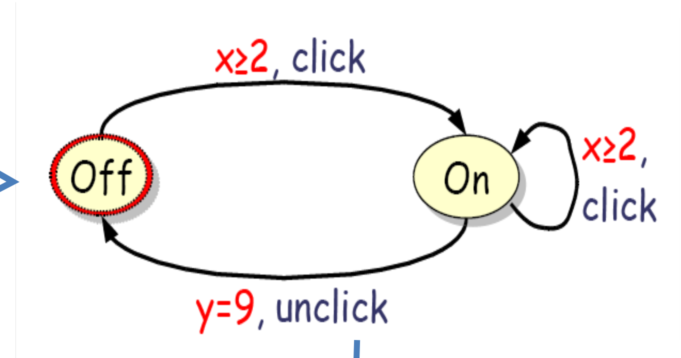
- Synchronisation Action transition : $(l\ ,u) \xrightarrow{a?a!} (l[l'_j\ /l_j, l'_i\ /l_i], u')$ if there exist $l_i \xrightarrow{g_i, a?, r_i} l'_i$ and $l_j \xrightarrow{g_j, a!, r_j} l'_j$ such that :

$u \in (g_i \wedge g_j)$, $u' = [r_i \cup r_j \rightarrow 0]u$ and $u' \in Inv(l\ [l'_j\ /l_j, l'_i/l_i])$.

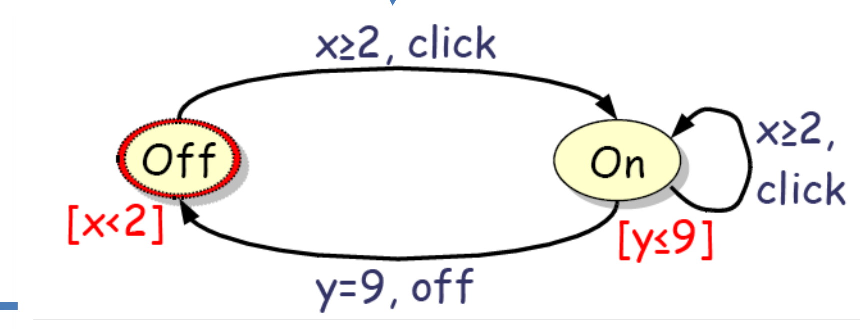
Example 1



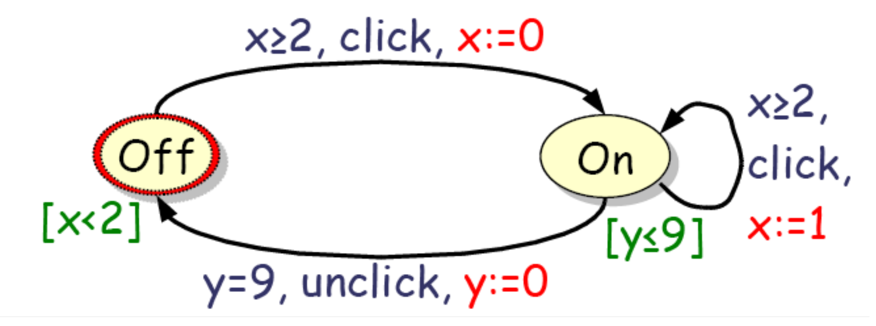
+ guards



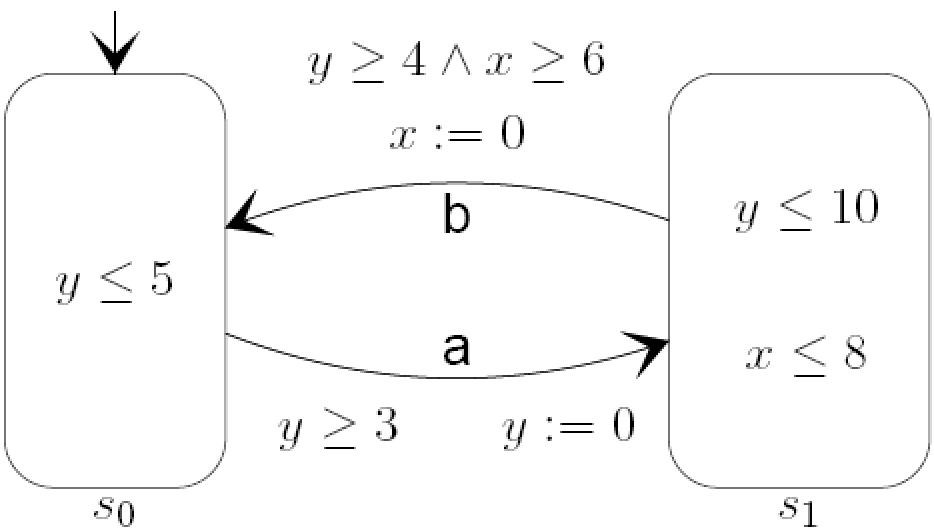
+ invariant



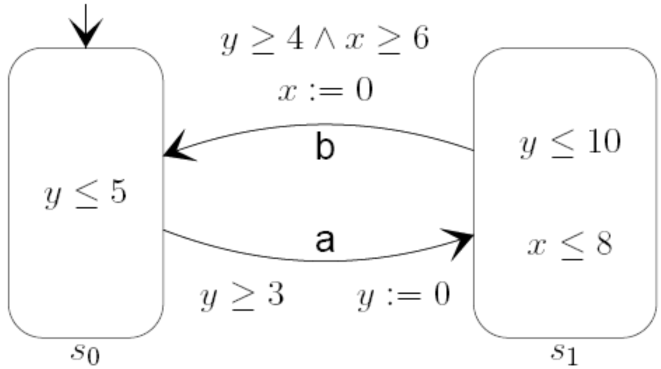
+ resets and updates



Example 2



Example 2: execution



(l,u)=(S_i, x, y)

$$(s0,0,0) \longrightarrow^{3} (s0,3,3) \longrightarrow^{a} (s1,3,0) \longrightarrow^{4} (s1,7,4) \longrightarrow^{b} (s0,0,4) \longrightarrow^{1} (s0,1,5)$$

$$\longrightarrow^{a} (s1,1,0) \ldots$$