Higher Dimensional Timed Automata

Emily Clement

6th of December 2023

Joint work with

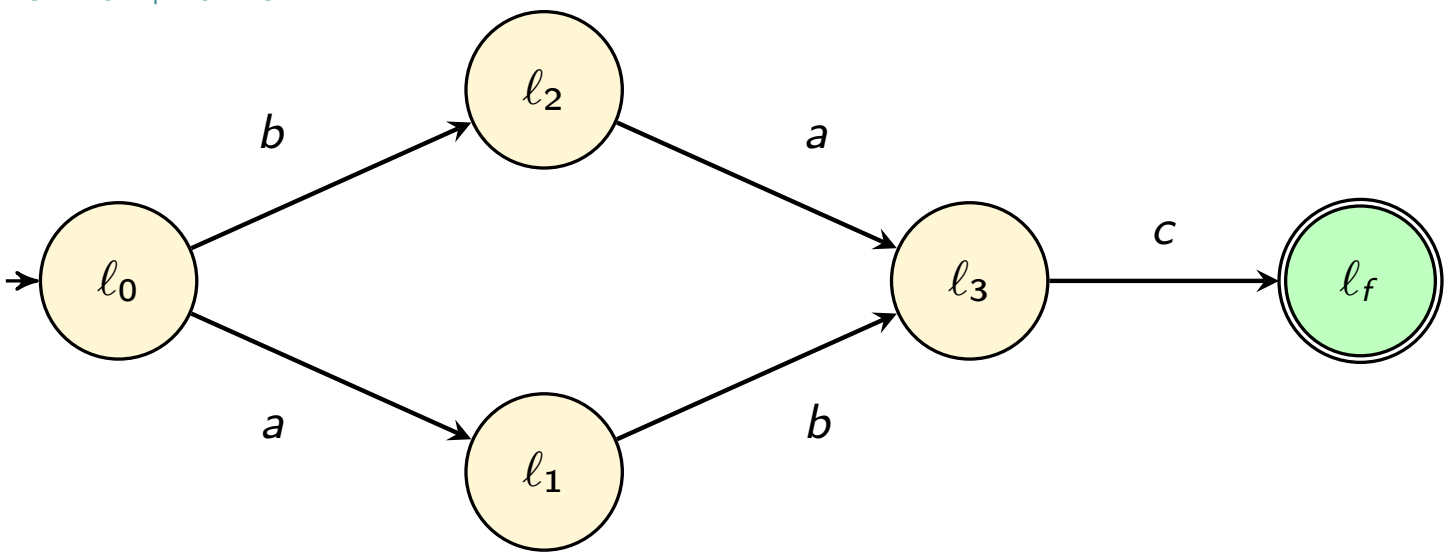
Uli Fahrbenberg

Amazigh Amrane

Hugo Bazille

Concurrency in Automata: interleaving concurrency

- $a \cdot b + b \cdot a$:



Higher Dimensional Automata: represent $n-$ary parallel events

- Goal of HDA

  Represent several events $a, b, c \cdots$ and their order

Higher Dimensional Automata: represent $n-$ary parallel events

- Goal of HDA

  Represent several events $a, b, c \cdots$ and their order

  Simple case: a and b can occur in parallel

Higher Dimensional Automata: represent $n-$ary parallel events

- Goal of HDA

  Represent several events $a, b, c \cdots$ and their order

  Simple case: a and b can occur in parallel

  Or.. a can **begin** before b does and a can **end** before b does ...

Higher Dimensional Automata: represent $n-$ary parallel events

- Goal of HDA

  Represent several events $a, b, c \cdots$ and their order

  Simple case: a and b can occur in parallel

  Or.. a can **begin** before b does and a can **end** before b does ...

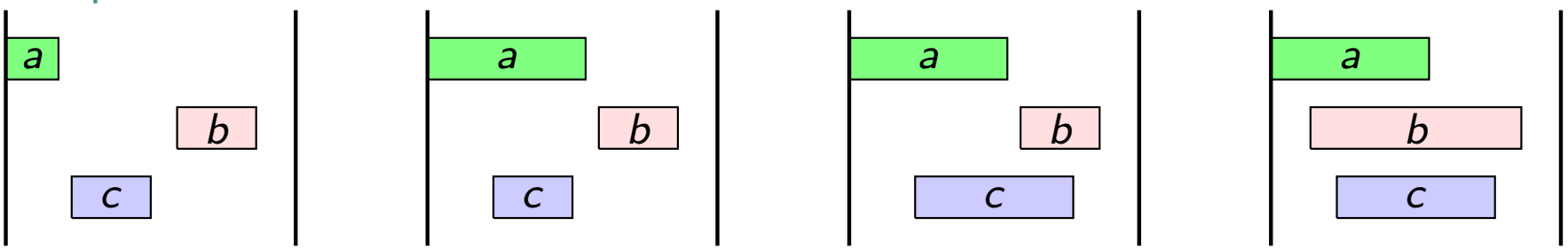
  While c happens before the end of b and a !

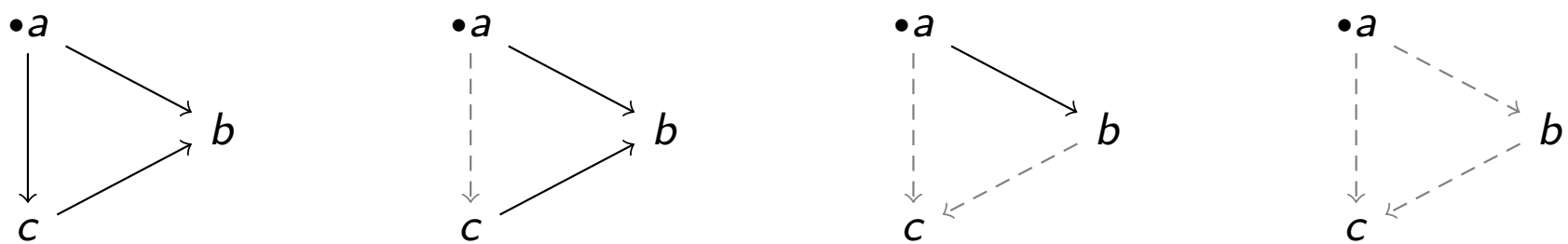
How to represent $n-$ary parallel events? Ipomset

- Two partial order events
  - $<$: precedence order (rep with $\longrightarrow$)
  - $\dashrightarrow$: event order.
  - $< \cup \dashrightarrow$: **total**.
- Interfaces

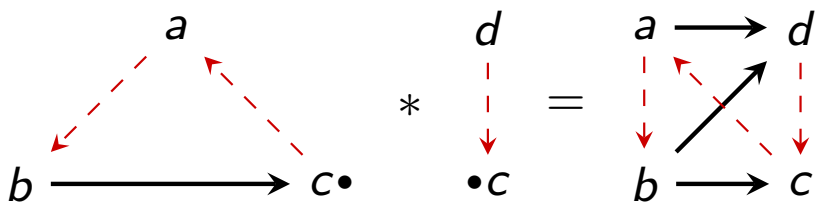
Source/Target interfaces: S/T: $<-$minimal/maximal.

- Representation of events as interval

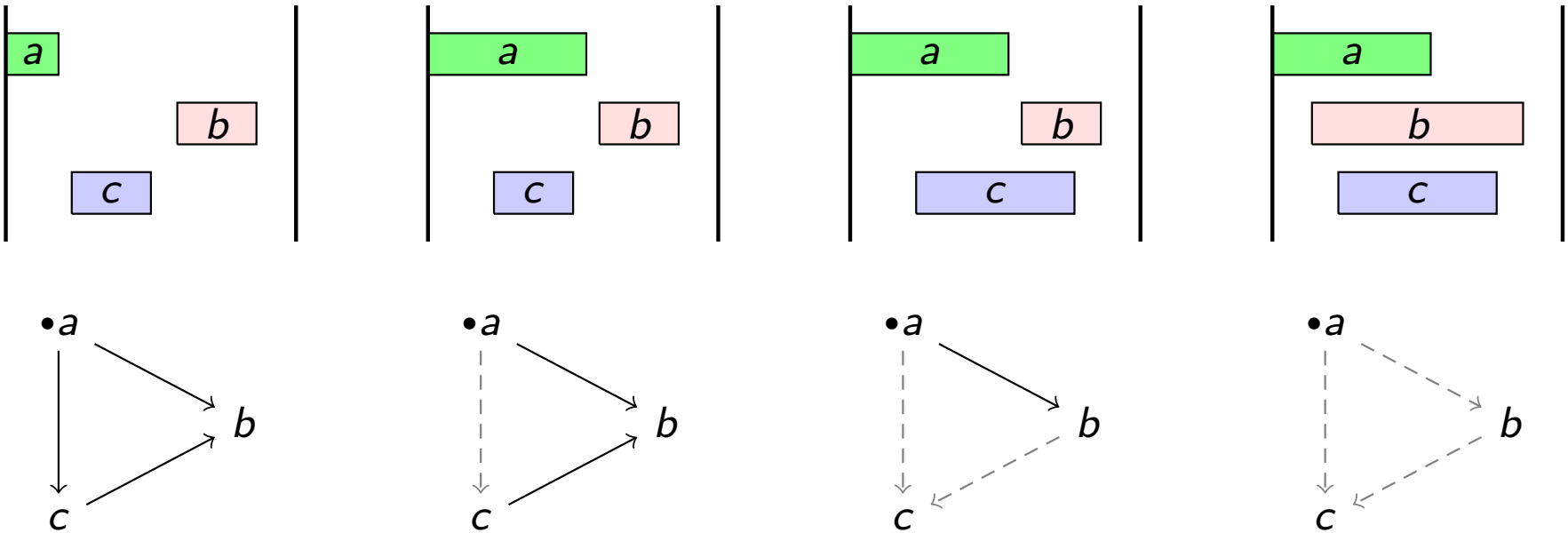




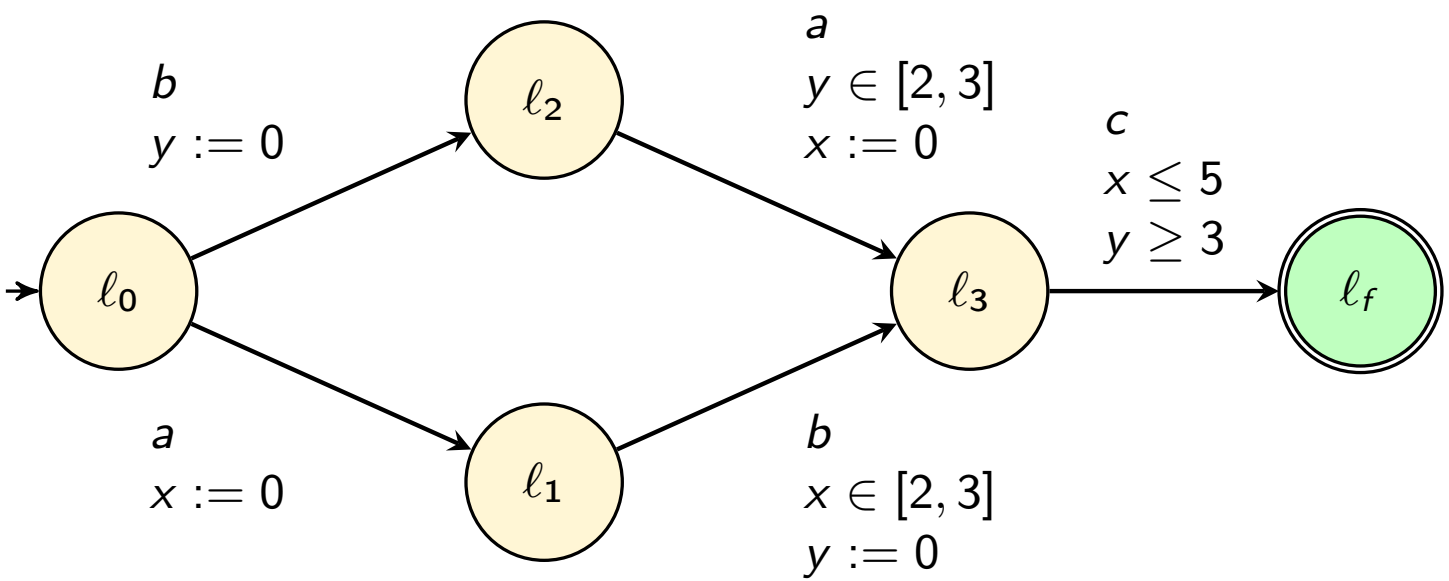
Example of operation: gluing



What about the time duration of events?



Interleaving concurrency with Timed Automata



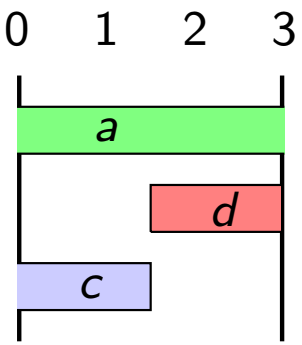
Interest in Higher Dimensional Timed Automata

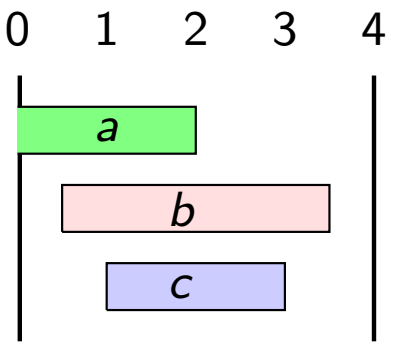
- HDA:
  - Representation concurrency relation
  - Extension of some properties of classic Automata
- Issues:
  - No information about time duration of each events
  - How to represent timing perturbation ?

Our goal

- Goals:

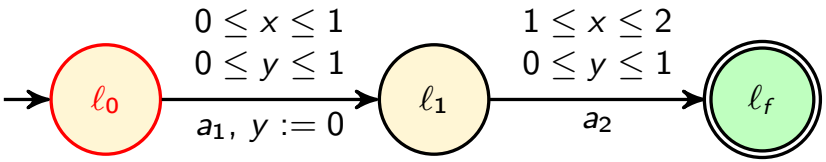
Extend the objects/operation/results from HDA to HDTA: adding information on the start/end dates of each event.



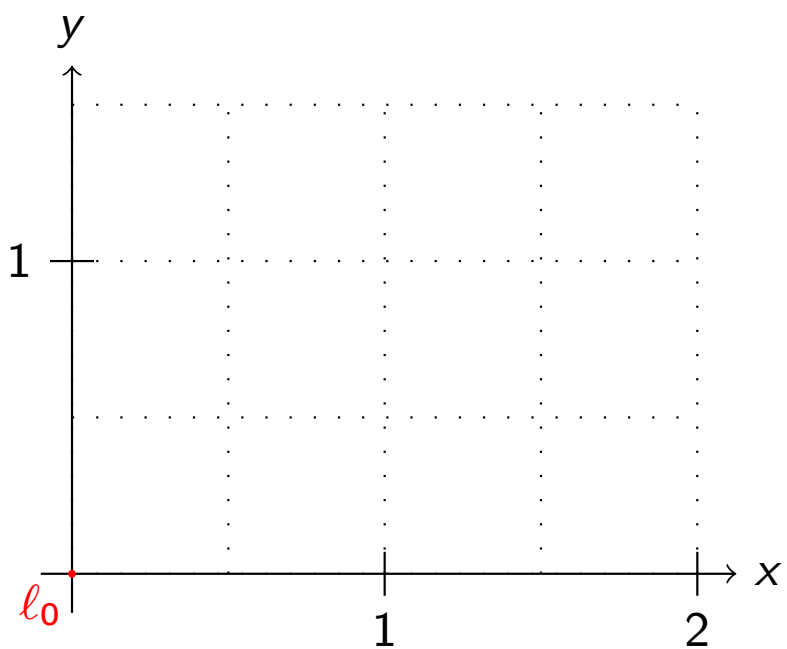


What about the robustness?

- Timed automaton $\mathcal{A}$:

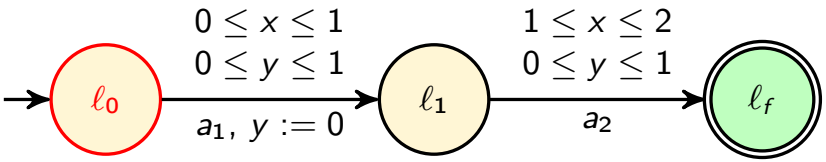


- Run with delay perturbations of at most $\delta = 0.2$

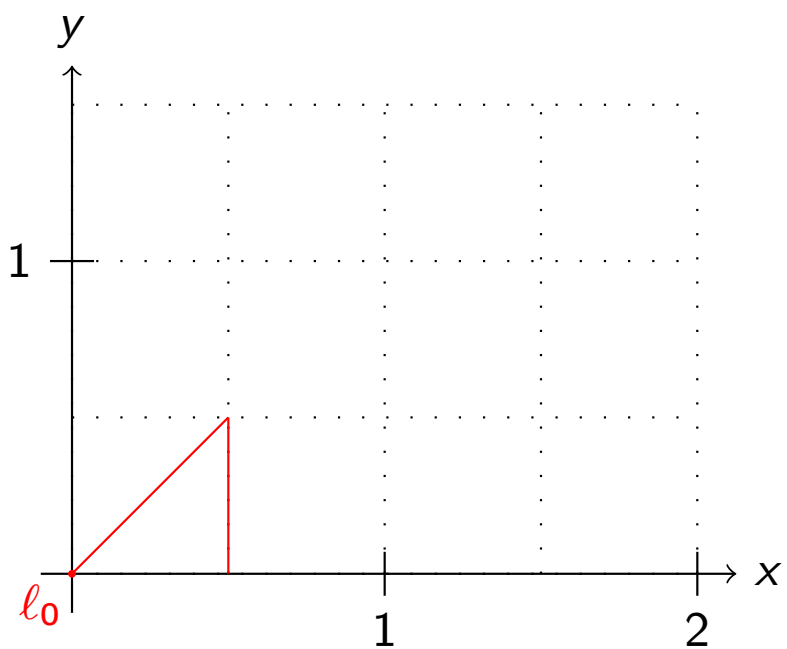


What about the robustness?

- Timed automaton $\mathcal{A}$:

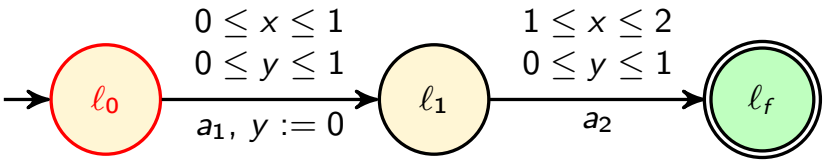


- Run with delay perturbations of at most $\delta = 0.2$

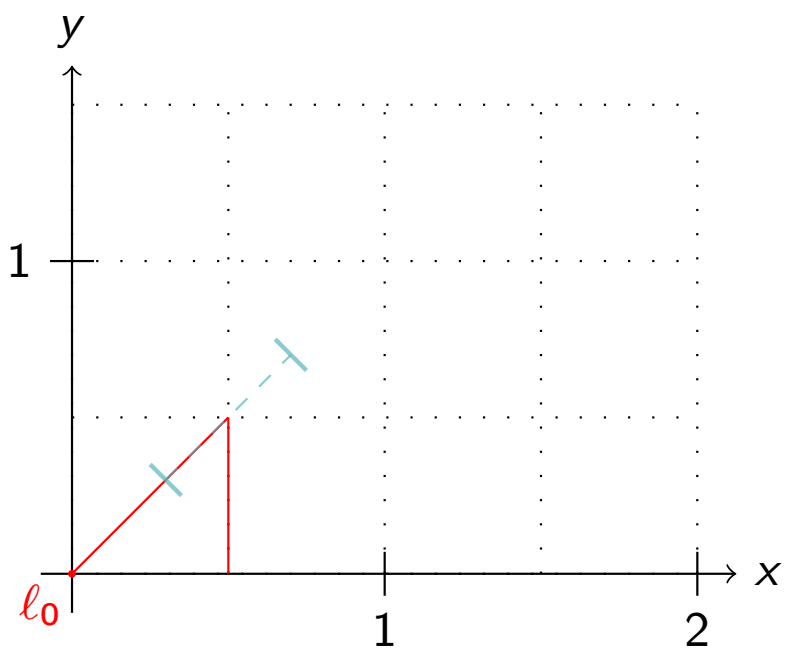


What about the robustness?

- Timed automaton $\mathcal{A}$:

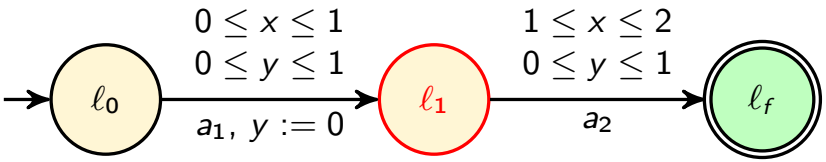


- Run with delay perturbations of at most $\delta = 0.2$

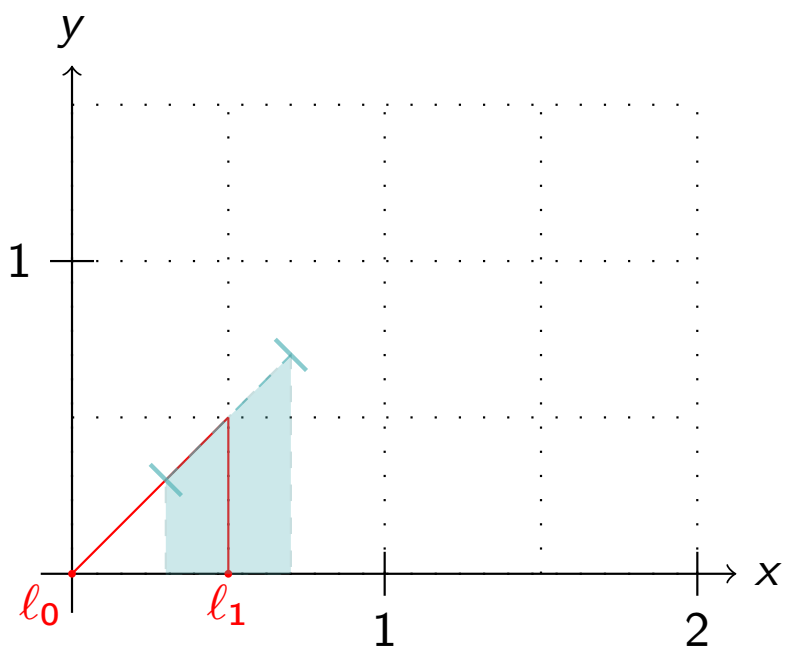


What about the robustness?

- Timed automaton $\mathcal{A}$:

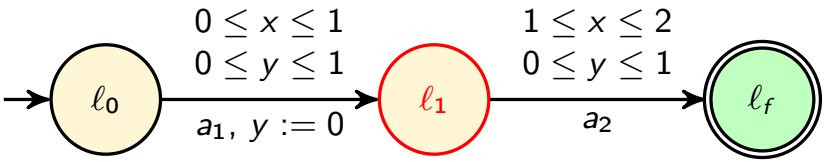


- Run with delay perturbations of at most $\delta = 0.2$

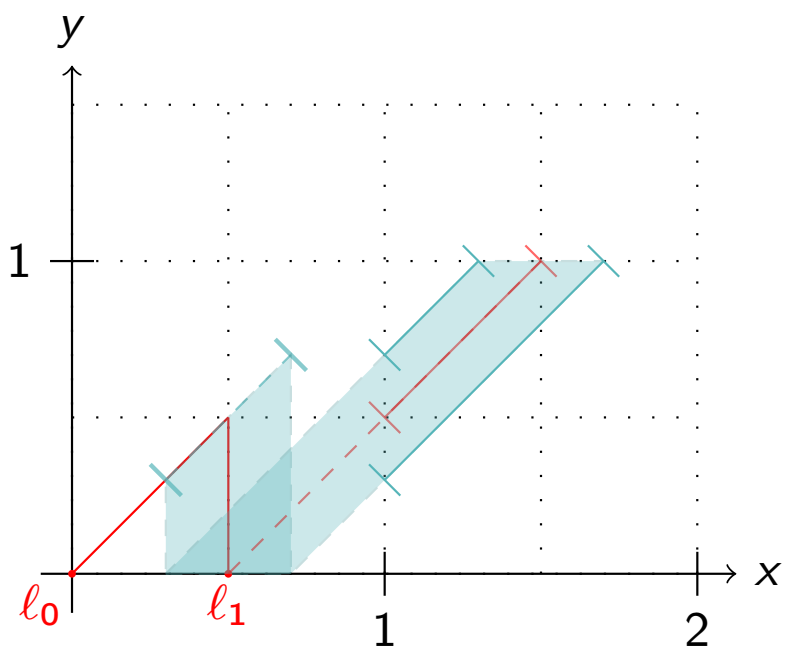


What about the robustness?

- Timed automaton $\mathcal{A}$:

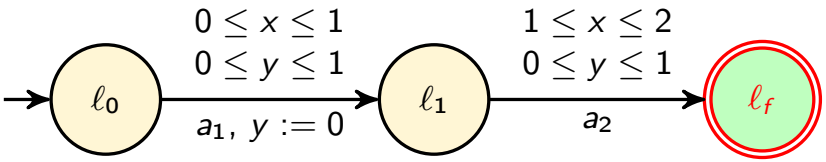


- Run with delay perturbations of at most $\delta = 0.2$

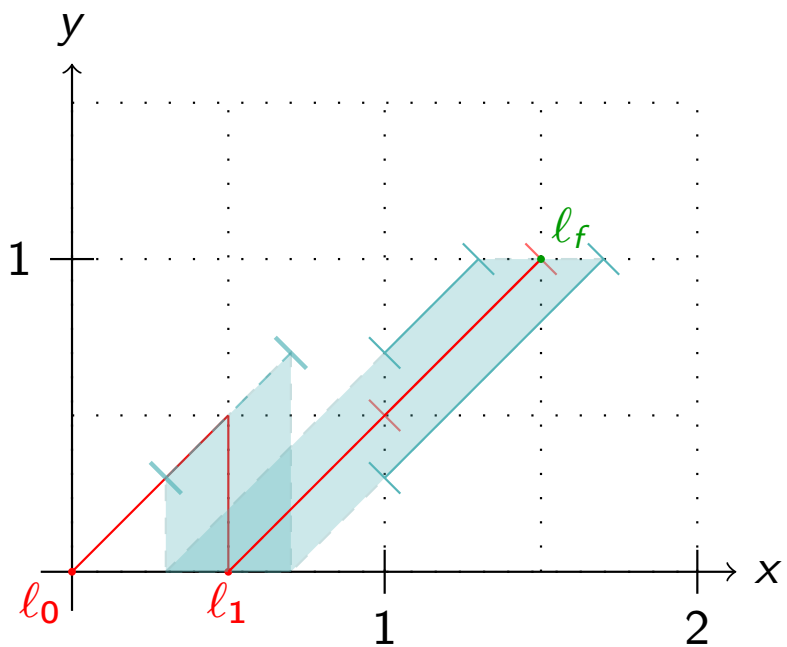


What about the robustness?

- Timed automaton $\mathcal{A}$:

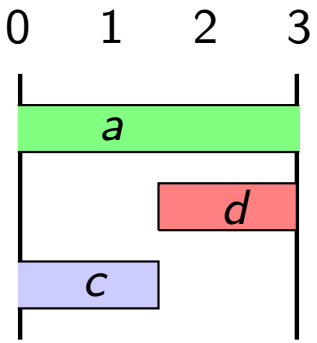
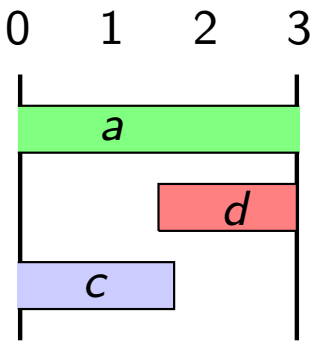


- Run with delay perturbations of at most $\delta = 0.2$



A future work: robustness on HDTA

- No timing perturbation: c and d are not in concurrency



- timing perturbation. Let us introduce a 0.1 delay on the end date of c: